MCR-85-641
NAS8-35922
PHASE TWO

(NASA-CR-179136) EXPERT SYSTEMS APPLIED TO FAULT ISOLATION AND ENERGY STORAGE MANAGEMENT, PHASE 2 Final Report (Martin Marietta Aerospace) 386 p Avail: NTIS HC A17/MF A01 CSCL 09B G3/61 N87-26530 Unclas 0085375

EXPERT SYSTEMS APPLIED TO FAULT ISOLATION AND ENERGY STORAGE MANAGEMENT

FINAL REPORT

Prepared for: National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Marshall Space Flight Center, AL 35812

TABLE OF CONTENTS

		Page
Chapter 1	Introduction	1
Chapter 2	Tutorial: A Simple Example	5
Chapter 3	System Versions	10
Chapter 4	LISP Interface Options	19
Chapter 5	Installation Guide	23
Chapter 6	Operating Constraints	26
Chapter 7	Battery Maintenance and Recharging	34
Chapter 8	Test Configuration Utility	41

APPENDICES

A. Operation Flowcharts and Screens

B. Test and Demonstration Plan

C. Software Design

D. Expert System Rules with Translation

E. Hardware Description and Drawings

CHAPTER 1

INTRODUCTION

INTRODUCTION

This manual serves as a user guide for the FIES II system. This includes a brief discussion of the background and scope of the project, a discussion of basic and advanced operating installation and problem determination procedures for the FIES II system and information on hardware and software design and implementation. The final part of the document consists of a number of appendices including a detailed specification for the microprocessor software, a detailed description of the expert system rule base and a description and listings of the LISP interface software.

Martin Marietta has expended considerable effort researching and developing AI technologies appropriate for the automated management of spacecraft power systems.

The first FIES (Fault Isolation Expert System) program, conducted under IR&D funding, demonstrated the applicability of expert systems to fault isolation and detection in space-based power systems.

The EMES (Energy Management Expert System) program, conducted under CR&D funding from MSFC, demonstrated that an expert planning system is capable of load management in spacecraft power distribution systems under both normal and degrading conditions.

While these projects provided proof of the concept that expert systems technology may play an important role in power system automation, neither of the above systems addressed the critical issues of performance of the expert systems themselves or the continuous operation that are key requirements for expert systems that play a role in an actual energy management system that integrates both traditional hardware and software based control strategies as well as expert systems based control strategies.

The FIES II program, conducted under CR&D funding from MSFC, has begun to address these requirements.

For FIES II, a breadboard which models the essential power components in a space based power system was constructed. The breadboard allowed MMC to drive a FIES system in a real time, hardware based simulation environment.

The following describes the hardware and software used for FIES II.

FIES hardware is divided into three major subsystems. The first is the Breadboard which contains solar array simulators (power supply modules), batteries, network configuration relays, load simulation modules, measurement networks, and control logic. These components may be configured into a power subsystem (also referred to as a power network) manually at the Bread Board Fault Insertion Panel, via keyboard entry at the Microprocessor Terminal, or through the Graphics/Mouse Interface on the SYMBOLICS Terminal. Four fault types are supported and may be inserted via the Bread Board Front Panel Fault Insertion switches. The fault types include:

`opened relay'	-	which interrupts current through some point in the power subsystem,
`closed relay'	-	which allows fault current to flow to some point in the power subsystem,
`resistive shunt'	-	which causes excessive current drain (1-2 amperes) at some point in the power subsystem, and
`direct shunt'	-	which causes an infinite current sink at some point in the power subsystem that is limited only by the current sourcing capability of the power modules.

The second major hardware subsystem is the Microprocessor. The Sensor Data Acquisition Schedule (SDAS) which is located on the microprocessor provides an operating system for managing and monitoring the Bread Board Subsystem. A bit oriented memory map allows individual control over the network configuration relays. Writing a `1' to the corresponding memory bit forces a relay contact closure and writing a `0' forces the relay contacts open. In this way, various power networks can be dynamically configured for evaluation and demonstration of the fault isolating capabilities of FIES.

The Microprocessor provides hardware for monitoring the voltage and current levels at 24 sensor node locations in the power subsystem. It is also equipped with added I/O capabilities for interfacing to the Bread Board Front Panel, the Monitor Terminal, and the SYMBOLICS machine.

The third hardware subsystem is the SYMBOLICS LISP machine. The SYMBOLICS provides the processing environment for the Expert System software, which is an extension of the Automated Reasoning Tool (ART). Graphic capabilities supported by the SYMBOLICS terminal provides a user environment that is easy to master.

The FIES II software is divided into three main subsystems (or layers). The layer closest to the Bread Board is the Sensor Data Acquisition Scheduler (SDAS) whose task is to coordinate the activities of the microprocessor. These activities include controlling the Bread Board hardware, monitoring the sensor measurement nodes for steady or fault states, and handling communications with the SYMBOLICS.

The middle layer, the Expert System Communications Interface, resides on the SYMBOLICS. Its activities include communicating with the microprocessor, supervising the sensor load network configuration, managing the sensor load measurement data, interacting with human operators, and interfacing to the Expert System.

The outermost layer is the Expert System itself. Its activities include interacting with the Expert System Interface and hypothesizing the source of faults manually introduced via the Fault Insertion Panel on the Bread Board.

CHAPTER 2

TUTORIAL: A SIMPLE EXAMPLE

TUTORIAL: A SIMPLE EXAMPLE

2.0 Overview

This section provides a programmed instruction approach to operating the FIES II system. A new user should be able to successfully execute all the instructions in the sequence prior to attempting to use the more sophisticated options provided by the system. Upon completion of the instructions, the user should have a basic understanding of how to operate the FIES II system. This tutorial is presented in 3 phases, hardware initialization, software initialization, and runtime. For each of the phases there is a flow chart in Appendix A keyed to the following text. Where appropriate, nodes in the flow chart reference graphic displays that are copies of what an operation should produce at each step in the sequence.

2.1 Hardware Initialization

Steps 2.1.1 thru 2.1.4 are concerned with correct initialization of the hardware.

2.1.1 Power up the breadboard. This is accomplished by throwing the main circuit breaker on the breadboard front panel to the "ON" position.

2.1.2 Ensure that all faults have been removed from the network. This implies that none of the 48 relay switches are toggled up or down. Ensure that the operating constraints detailed in chapter 6 are in effect.

2.1.3 Verify that the microprocessor operating system has completed self test. Operating system load and self test will occur automatically when the breadboard is powered on. Successful completion is verified by the presence of a '-' prompt on the microprocessor terminal.

2.1.4 Boot the Symbolics processor. This is accomplished by entering: logout ‹return› then: halt machine ‹return›. At this time the system will prompt for input. Enter boot art.boot ‹RETURN›. Booting the system will take about 3 minutes. Verify that the boot sequence has completed by observing the "please login message" on the terminal.

2.2 Software Initialization

Steps 2.2.1 thru 2.2.9 are concerned with correct initialization of system and application software.

2.2.1 Enter "SDAS.EXE" at the microprocessor terminal. The system should not respond in any way.

2.2.2 Turn the switch on the front panel of the breadboard to self test. This should result in a sequence of relay led configurations displayed on the breadboard.

2.2.3 After observing the "self test completed" message on the breadboard status display, return the front panel switch to reset.

2.2.4 After observing the "system has been reset" message on the breadboard status display, turn the front panel switch to slave mode. Verify that the status display reads "SYMBOLICS MODE".

2.2.5 At the Symbolics terminal enter (login 'fies-production). This will cause the interface software to be loaded. If the "MORE" message appears at the bottom of the terminal, simply depress the space bar.

2.2.6 Enter ART by depressing the SELECT key, waiting 2-3 seconds and then entering A. Verify that you are in the ART command window, more recent versions of ART go to the next step (clear) automatically.

2.2.7 In response to the = > prompt in the ART command window, enter "clear". Verify that the system clears. If no prompt is visible after a clear, depress ‹RETURN›.

2.2.8 In response to the = > prompt in the ART command window, enter "load". The system will respond by prompting for a file name. Enter c: > fies-art-production> load-fies.lisp. Observe that the entire system is loaded by watching that rules compile as they are loaded. This will take about 5 minutes. Upon completion the system should display the = > prompt.

2.2.9 In response to the = > prompt enter reset. The system will initialize the ART schemata and database. This will take about 3 minutes.

2.3.0 Initiate Runtime

Sections 2.3.1 thru 2.3.10 describe the steps taken to run a fault isolation case.

2.3.1 In response to the = > prompt in the ART command window enter "run". The system should transfer control to the LISP interface schematic display.

2.3.2 Position the mouse to the middle of the display and click once on the middle button. This should result in a pop up menu.

2.3.3 Position the crosshair cursor to the RETRIEVE CONFIGURATION option in the menu. Observe that a box appears around the option. Now click on the left mouse button.

2.3.4 A window should now appear at the top of the screen. In response to the "enter file name" prompt type in msfc-demo-1.cnf and hit return.

2.3.5 Observe that the configuration stored in the above file has been loaded into the LISP interface schematic display.

2.3.6 Click on the middle mouse button, resulting in the pop up menu. Now select the download configuration option and click left on the mouse.

2.3.7 Watch the Fies interface status display. When the message in the display reads "I/O wait: fault data" you may insert a fault in the breadboard.

2.3.8 Ensure that the second shunt type switch is toggled down to direct. Then toggle the relay numbered 23 down. This introduces a direct short on the path leading from power module 2 to the low power bus.

2.3.9 The microprocessor will then transfer the faulted state to the LISP machine. Following this data transfer, control will be passed to ART. ART will run the FIES application. Typical times for this will be between 1.5 and 4 minutes.

2.4 Repeated Runs

It is not necessary to complete the entire sequence described above for every fault isolation run. Once the ART application has been loaded (step 2.2.8) there is no need to reload it. The procedure for a repeated run is to begin with step 2.2.9, continuing through section 2.3.9 until completion.

CHAPTER 3

SYSTEM VERSIONS

SYSTEM VERSIONS

3.0 Overview

MMC has provided two FIES II systems. The first of these systems is called the production version. This system operates in near-real time mode. For this reason, the production version offers no explanation facility beyond that of the screen that displays fault types and locations. The second of these systems is called the interactive system. Here the user can examine the behavior of the LISP interface and the sort runtime environment through a variety of switches available through both LISP and ART.

3.1 Production Version

The operating instructions for the production version appear below. These steps are represented graphically by the flowchart for the production version which appears in Appendix A.

3.1.1 Hardware Initialization

Steps 3.1.1 thru 3.1.4 are concerned with correct initialization of the hardware.

3.1.1.1 Power up the breadboard. This is accomplished by throwing the main circuit breaker on the breadboard front panel to the "ON" position.

3.1.1.2 Ensure that all faults have been removed from the network. This implies that none of the 48 relay switches are toggled up or down. Ensure that the operating constraints detailed in chapter 6 are in effect.

3.1.1.3 Verify that the microprocessor operating system has completed self test. Operating system load and self test will occur automatically when the breadboard is powered on. Successful completion is verified by the presence of a '-' prompt on the microprocessor terminal.

3.1.1.4 Boot the Symbolics processor. This is accomplished by entering: logout ‹return› then: machine halt ‹return›. At this time the system will prompt for input. Enter boot art.boot ‹RETURN›. Booting the system will take about 3 minutes. Verify that the boot sequence has completed by observing the "please login message" on the terminal.

3.1.2 Software Initialization

Steps 3.1.2.1 thru 3.1.2.9 are concerned with correct initialization of system and application software.

3.1.2.1 Enter "SDAS.EXE" at the microprocessor terminal. The system should not respond in any way.

3.1.2.2 Turn the switch on the front panel of the breadboard to self test. This should result in a sequence of relay leds configurations displayed on the breadboard.

3.1.2.3 After observing the "self test completed" message on the breadboard status display, return the front panel switch to reset.

3.1.2.4 After observing the "system has been reset" message on the breadboard status display, turn the front panel switch to slave mode. Verify that the status display reads "SYMBOLICS MODE".

3.1.2.5 At the symbolics terminal enter (login 'fies production). This will cause the interface software to be loaded. If the *MORE* message appears at the bottom of the terminal, simply depress the space bar.

3.1.2.6 Enter ART by depressing the SELECT key, waiting 2-3 seconds and then entering A. Verify that you are in the ART command window more recent versions of ART go to the next step (clear) automatically.

3.1.2.7 In response to the = > prompt in the ART command window, enter "clear". Verify that the system clears. If no prompt is visible after a clear, depress <RETURN>.

3.1.2.8 In response to the = > prompt in the ART command window, enter "load". The system will respond by prompting for a file name. Enter the string C: > fies-art-production> load-fies.lisp. Observe that the entire system is loaded by watching that rules compile as they are loaded. This will take about 5 minutes. Upon completion the system should display the = > prompt.

3.1.2.9 In response to the = > prompt enter reset. The system will initialize the ART schemata and database. This will take about 3 minutes.

3.1.3 Initiate Runtime

Sections 3.1.3.1 thru 3.1.3.9 describe the steps taken to run a fault isolation case.

3.1.3.1 In response to the = > prompt in the ART command window enter "run". The system should transfer control to the LISP interface schematic display.

3.1.3.2 Position the mouse to the middle of the display and click once on the middle button. This should result in a pop up menu.

3.1.3.3 Position the crosshair cursor to the RETRIEVE CONFIGURATION option in the menu. Observe that a box appears around the option. Now click on the left mouse button. Alternatively the user may enter a configuration by mousing on the desired relays. If this is done the user may skip step 3.1.3.4.

3.1.3.4 A window should now appear at the top of the screen. In response to the "enter file name" prompt type in the name of the desired configuration file.

3.1.3.5 Observe that the configuration stored in the above file has been loaded into the LISP interface schematic display.

3.1.3.6 Click on the middle mouse button, resulting in the pop up menu. Now select the download configuration option and click left on the mouse.

3.1.3.7 Watch the Fies interface status display. When the message in the display reads "I/O wait: fault data" you may insert a fault in the breadboard

3.1.3.8 Insert the required fault in breadboard.

3.1.3.9 The microprocessor will then transfer the faulted state to the LISP machine. Following this data transfer, control will be passed to ART. ART will run the FIES application. Typical times for this will be between 1.5 and 4 minutes.

3.1.4 Repeated Runs

It is not necessary to complete the entire sequence a described above for every fault isolation run. Once the ART application has been loaded (Step 3.1.2.8) there is no need to reload it. The procedure for a repeated run is to begin with step 3.1.2.9, continuing through section 3.1.3.9 until completion.

3.2 Interactive Version

Operation for the interactive model is described in the following section. This version allows the operator to step through the runtime system, examining steady state and faulted data, as they are received by the LISP interface, setting up ART runtime options and finally controlling the execution of the ART application.

In general, the operation is very similar to the operations described in the tutorial. The major differences are the directory from which the applications is loaded and the availability of ART options during runtimes.

The operating instruction for the interactive version appear below. These steps are represented graphically by the flowchart for the interactive version in Appendix A.

3.2.1 Hardware Initialization

Steps 3.2.1.1 thru 3.2.1.4 are concerned with correct initialization of the hardware.

3.2.1.1 Power up the breadboard. This is accomplished by throwing the main circuit breaker on the breadboard front panel to the "ON" position.

3.2.1.2 Ensure that all faults have been removed from the network. This implies that none of the 48 relay switches are toggled up or down. Ensure that the operating constraints detailed in chapter 6 are in effect.

3.2.1.3 Verify that the microprocessor operating system has completed self test. Operating system load and self test will occur automatically when the breadboard is powered on. Successful completion is verified by the presence of a "-' prompt on the microprocessor terminal.

3.2.1.4 Boot the symbolics processor. This is accomplished by entering: logout ‹return› then: halt machine. At this time the system will prompt for input. Enter boot art.boot ‹RETURN›. Booting the system will take about 3 minutes. Verify that the boot sequence has completed by observing the "please login message" on the terminal.

3.2.2 Software Initialization

Steps 3.2.2.1 thru 3.2.2.9 are concerned with correct initialization of system and application software.

3.2.2.1 Enter "SDAS.EXE" at the microprocessor terminal. The system should not respond in any way.

3.2.2.2 Turn the switch on the front panel of the breadboard to self test. This should result in a sequence of relay leds configurations displayed on the breadboard.

3.2.2.3 After observing the "self test completed" message on the breadboard status display, return the front panel switch to reset.

3.2.2.4 After observing the "system has been reset" message on the breadboard status display, turn the front panel switch to slave mode. Verify that the status display reads "SYMBOLICS MODE".

3.2.2.5 At the symbolics terminal enter (login 'fies interactive). This will cause the interface software to be loaded. If the *MORE* message appears at the bottom of the terminal, simply depress the space bar.

3.2.2.6 Enter ART by depressing the SELECT key, waiting 2-3 seconds and then entering A. Verify that you are in the ART command window (more recent versions of ART go to the next step (clear) automatically).

3.2.2.7 In response to the = > prompt in the ART command window, enter "clear". Verify that the system clears. If no prompt is visible after a clear, depress <RETURN>.

3.2.2.8 In response to the = > prompt in the ART command window, enter "load". The system will respond by prompting for a file name. Enter the string C: > fies-art-interactive> load-fies.lisp. Observe that the entire system is loaded by watching that rules compile as they are loaded. This will take about 5 minutes. Upon completion the system should display the = > prompt.

3.2.2.9 In response to the = > prompt enter reset. The system will initialize the ART schemata and database. This will take about 3 minutes.

3.2.3 Initiate Runtime

Sections 3.2.3.1 thru 3.2.3.10 describe the steps taken to run a fault isolation case.

3.2.3.1 In response to the = > prompt in the ART command window enter "run". The system should transfer control to the LISP interface schematic display.

3.2.3.2 Position the mouse to the middle of the display and click once on the middle button. This should result in a pop up menu.

3.2.3.3 Position the crosshair cursor to the RETRIEVE CONFIGURATION option in the menu. Observe that a box appears around the option. Now click on the left mouse button. Alternatively the user may enter a configuration by mousing on the desired relays. If this is done the user may skip step 3.3.4.

3.2.3.4 A window should now appear at the top of the screen. In response to the "enter file name" prompt type in the name of the file and hit return.

3.2.3.5 Observe that the configuration stored in the above file has been loaded into the LISP interface schematic display.

3.2.3.6 Click on the middle mouse button, resulting in the pop up menu. Now select the download configuration option and click left on the mouse. After the configuration has been loaded and the steady state table returned to the LISP interface, the user may examine the steady state data using the EXAMINE STEADY STATE menu option.

3.2.3.7 Watch the Fies interface status display. When the message in the display reads "I/O wait: fault data" you may insert a fault in the breadboard

3.2.3.8 Insert the required fault into the breadboard.

3.2.3.9 The microprocessor will then transfer the faulted state to the LISP machine. The user may then examine the faulted state, using the EXAMINE Breadboard data menu option.

3.2.3.10 The user may now set up the ART runtime environment. This is accomplished by clicking on "WATCH" in the ART ROOT MENU and selecting 1 or more of the rules, facts, goals, agenda, activations, statistics and dribble options. Then click twice on the middle mouse button to return to the ROOT menu. Now click on "RUN" to start the application.

3.2.4 Repeated Runs

It is not necessary to complete the entire sequence as described above for every fault isolation run. Once the ART application has been loaded (Step 3.2.2.8) there is no need to reload it. The procedure for a repeated run is the begin with step 3.2.2.9, continuing through section 3.2.3.10 until completion. If any of the WATCH options have been turned on, the user should turn them off prior to the reset command. Failure to do so will cause errors in the interface between LISP and ART.

CHAPTER 4

LISP INTERFACE OPTIONS

LISP INTERFACE OPTIONS

4.0 Overview

This section provides a description of the options available in the LISP interface. It is anticipated that these options will be primarily used in conjunction with the interactive version of the FIES system.

4.1 Structure

Options are provided through a menu system. The menu system is only one level deep, that is, the only option available to a user after selecting and executing an option from the main menu is to return to the main menu.

4.2 Options

The following options are available through the menu system. Each of the options is described below.

4.2.1 Download Configuration

This option is used to download a configuration to the microprocessor prior to a run. The configuration is specified either by using the RETRIEVE CONFIGURATION menu option to load a previously saved configuration into the display or by clicking the required relays off or on using the mouse.

4.2.2 Retrieve Configuration

This option is used to load a previously saved configuration into the display from the fies-library-directory. Upon clicking left on this menu item the user should observe a window at the top of the screen. The window may be exited by using the ABORT key.

Once the window is visible, type in the name of the file corresponding to the desired configuration. After the name is entered, hit <RETURN>. The window should

disappear and the user should observe that the configuration has been loaded into the display. If the file does not exist the message "file does not exist" will appear in the FIES Interface Status window.
At this point the user will be reprompted for the file name. Use the ABORT key to exit the window or reenter the file name. In order to view all saved configurations, the user should use the LISP machine file system. This is accomplished by the following steps:

1) SELECT F

2) Click left on TREE EDIT ANY Option

3) Enter c:fies > fies-library > *.*.*

The user should maintain this directory using commands native to the LISP environment. These commands are discussed in the ZMACS edit manual.

4.2.3 Archive Configuration

This option is used to store a configuration that is currently loaded in the display into a file in the fies-library directory.

After selecting this option, the user should observe a window at the top of the display. Enter the name of the file in which the configuration is to be stored and hit <RETURN>. If the file already exists the message "File exists" will appear in the FIES Interface status window. In this case, the user will be reprompted for a new file name. Enter the new file name or use the ABORT key to exit the window.

While any naming conventions may be used, it is suggested that a common extension such as ".cnf" be employed. This is the extension use for MMC provided configuration files. A directory name need not be entered, as the default will be to store the files in c: > fies > fies-library.

4.2.4 Examine Steady State Data

This option allows the user to examine the state of the breadboard following a download and the subsequent transmission of the steady-state table to the LISP machine. To return from the steady state window display, the user should click the middle mouse button and select the option "RETURN TO PREVIOUS SCREEN".

4.2.5 Examine Breadboard Data

This option may only be used in the interactive version of the system. The option allows the user to examine the state of the breadboard following fault insertion and the subsequent transmission of the fault table to the LISP machine. To return from the breadboard window display, the user should click the middle mouse button and select the option "RETURN TO PREVIOUS SCREEN".

4.2.6 Art Command Window

This option is only useful in the interactive version of the system. Typically, it is used following a configuration download steady state transmission, fault insertion and fault state transmission. It allows the user to transfer control to ART after reviewing the data (i.e. using examine steady state or examine breadboard data command options).

If the option is used in any other sequence, result of the ART run will be meaningless since ART is expecting a fault state data input.

4.2.7 QUIT

This option allows the user to exist the interface and return to the LISP LISTENER.

CHAPTER 5

INSTALLATION GUIDE

INSTALLATION

5.0 Overview

This section details the steps necessary to install the FIES software. Installation includes microprocessor, LISP machine interface and expert system software.

5.1 Microprocessor Installation

5.1.1 Ensure that the IRMX operating system has been correctly installed. This may be accomplished by booting the microprocessor and observing that the microprocessor self confidence test runs to completion.

5.1.2 Use the backup utility to load the FIES microprocessor software from the 8" floppy disk to the winchester.

5.1.3 Enter "SDAS.EXE" from the microprocessor console. If there is no response, the system is functioning normally.

5.1.4 Further microprocessor software verification test procedures may be found in the test and demonstration plan document.

5.2 Lisp Machine Interface Software Installation

5.2.1 Mount the cartridge tape containing the FIES system in the tape drive.

5.2.2 Create the following directories:
1) c: > fies
2) c: > fies > fies-library
4) c: > fies > fies-art-production
5) c: > fies > fies-art-interactive

5.2.3 From the Symbolics terminal, enter the file system editor by entering SELECT F.

5.2.4 Click the mouse left on Local File Maintenance option in the File System editor command window.

5.2.5 Click the mouse left on the Reload/Retrieve option in the command window.

5.2.6 Enter the following pathnames into the options window:
1) c: > fies > *.*.*
2) c: > fies > fies-library *.*.*
3) c: > fies > fies-test-library *.*.*
4) c: > fies > fies-art-production *.*.*
5) c: > fies > fies-art-interactive *.*.*

5.2.7 Click left on the Reload all option in the options window.

5.2.8 Click left on DO IT in the options window.

5.2.9 Observe that the files have been loaded into the appropriate directories

5.2.10 From the Lisp Listener, enter (logout).

5.2.11 From the Lisp Listener, enter (login 'fies). Observe that the FIES Interface software is loaded into the system.

5.2.12 Further LISP interface software verification procedures may be found in the test and demonstration plan document.

5.3 Expert System Installation

5.3.1 Installation of the expert system application is accomplished at the same time the LISP machine interface software is installed.

CHAPTER 6

OPERATING CONSTRAINTS

OPERATING CONSTRAINTS

6.0 This chapter details operational constraints for the breadboard. These conditions must be in effect at runtime for the conclusion of a successful demonstration. Section 6.4 presents a Breadboard Readiness checkout procedure.

6.1 Constraints

Sink - Source relationships

The sum of the amperages drawn by sinks (loads) in the system must be less than or equal to the sum of amperages supplied by sources (solar arrays and batteries in the system). Each solar array is capable of providing approximately 1.8 amps. A battery is considered to be capable of safely providing .5 amps to the system. Hence the configuration in example 6-1, which draws 3.74 amps is permissible, while the configuration in example 6-2, which draws 4.84 amps is not.

6.2 Single Battery Configuration

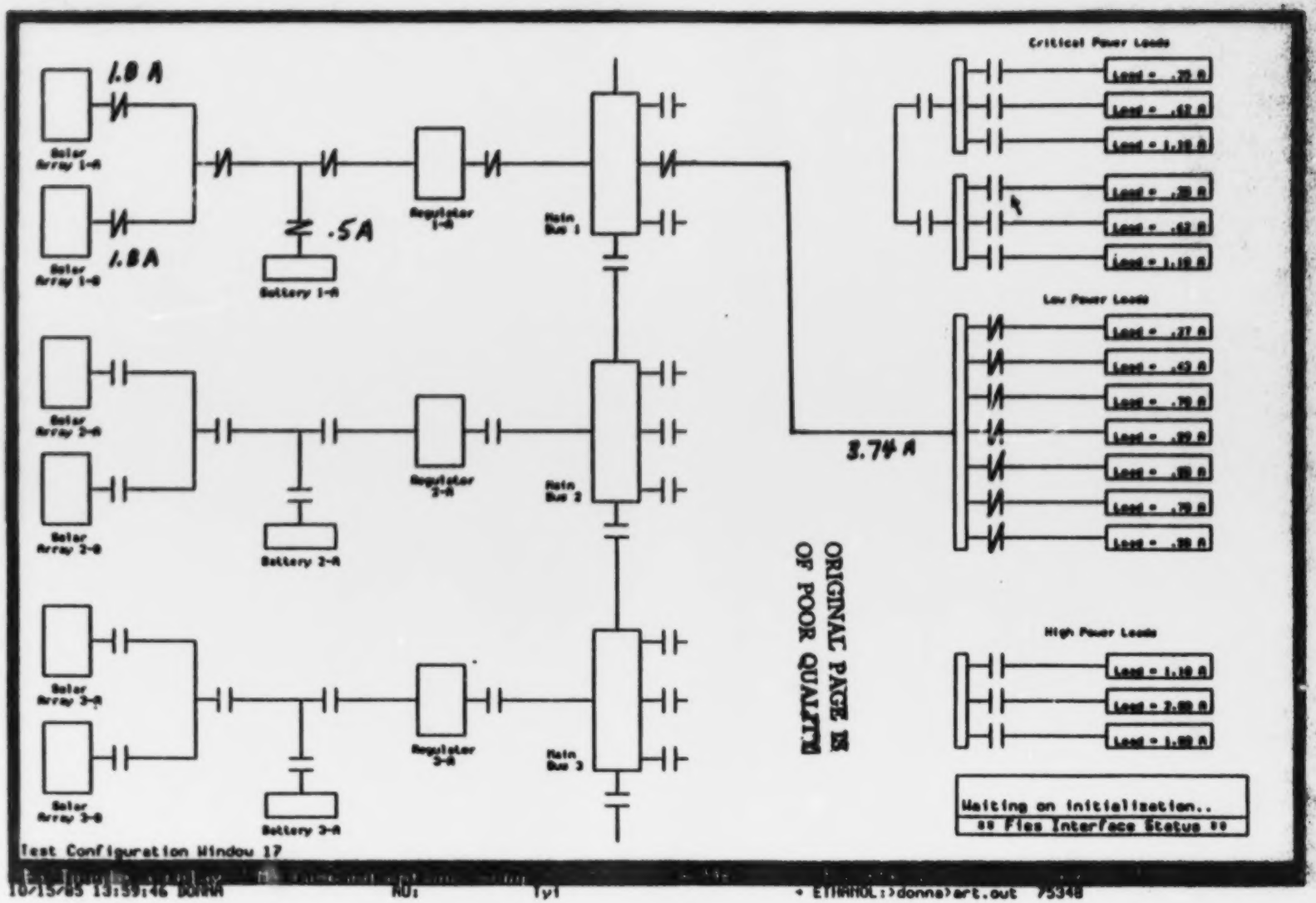
As stated in 6.1, no battery should drive loads requiring more than .5 amps

6.3 Battery Maintenance

Batteries should be fully charged prior to any demonstration of the system. Failure to do so will result in spurious faults in most configurations, resulting in incorrect diagnosis by the expert system.

Since there is no way to determine whether or not a battery is fully charged, all batteries should be trickle charged for 8 hours in advance of any demonstration. Using the test-configuration utility to download the battery charge configuration (Example 6-3) to the breadboard and allowing the breadboard to remain stable for at least 8 hours.

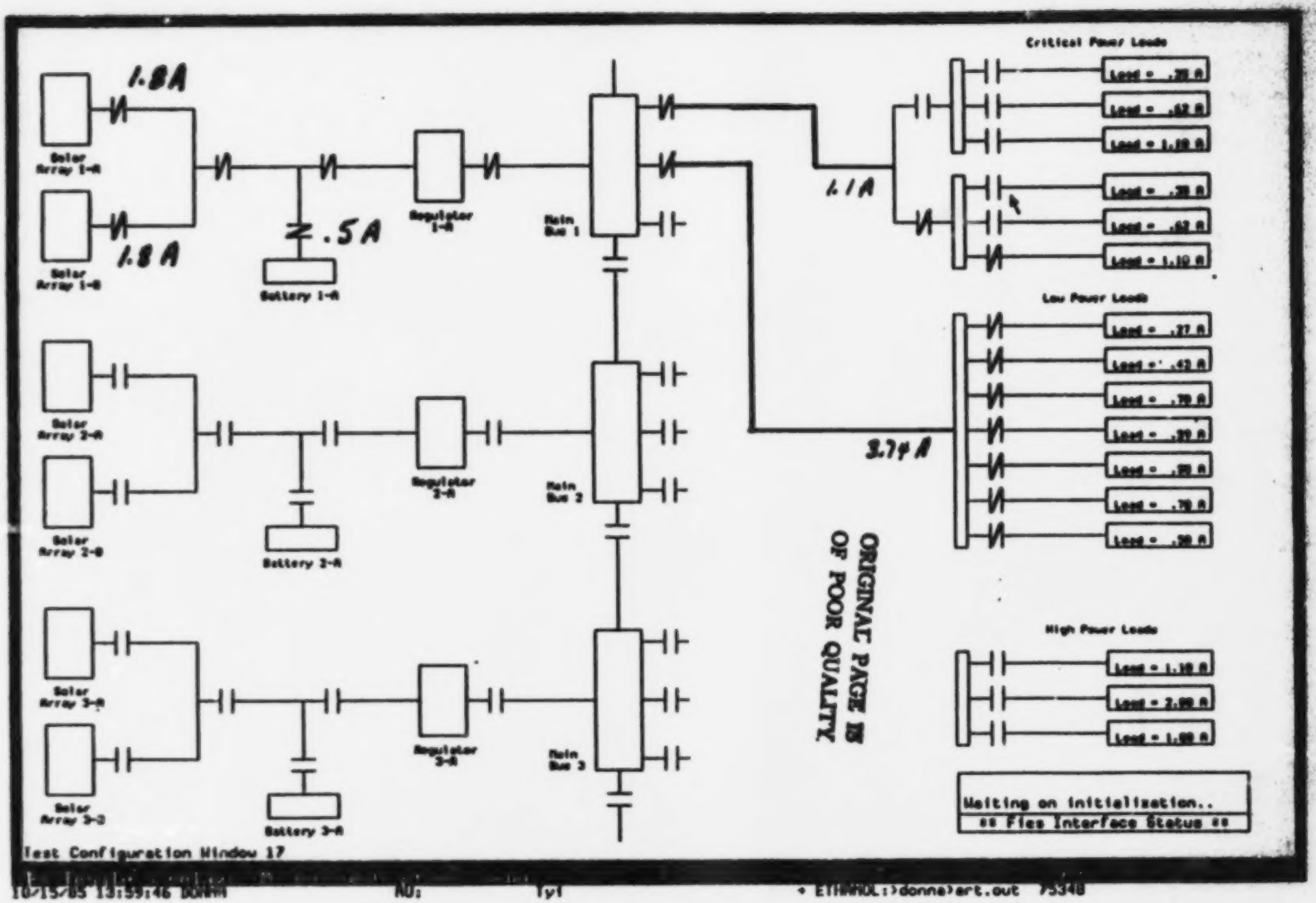
A detailed discussion of battery maintenance and recharging may be found in chapter 7.0.



ORIGINAL PAGE IS OF POOR QUALITY

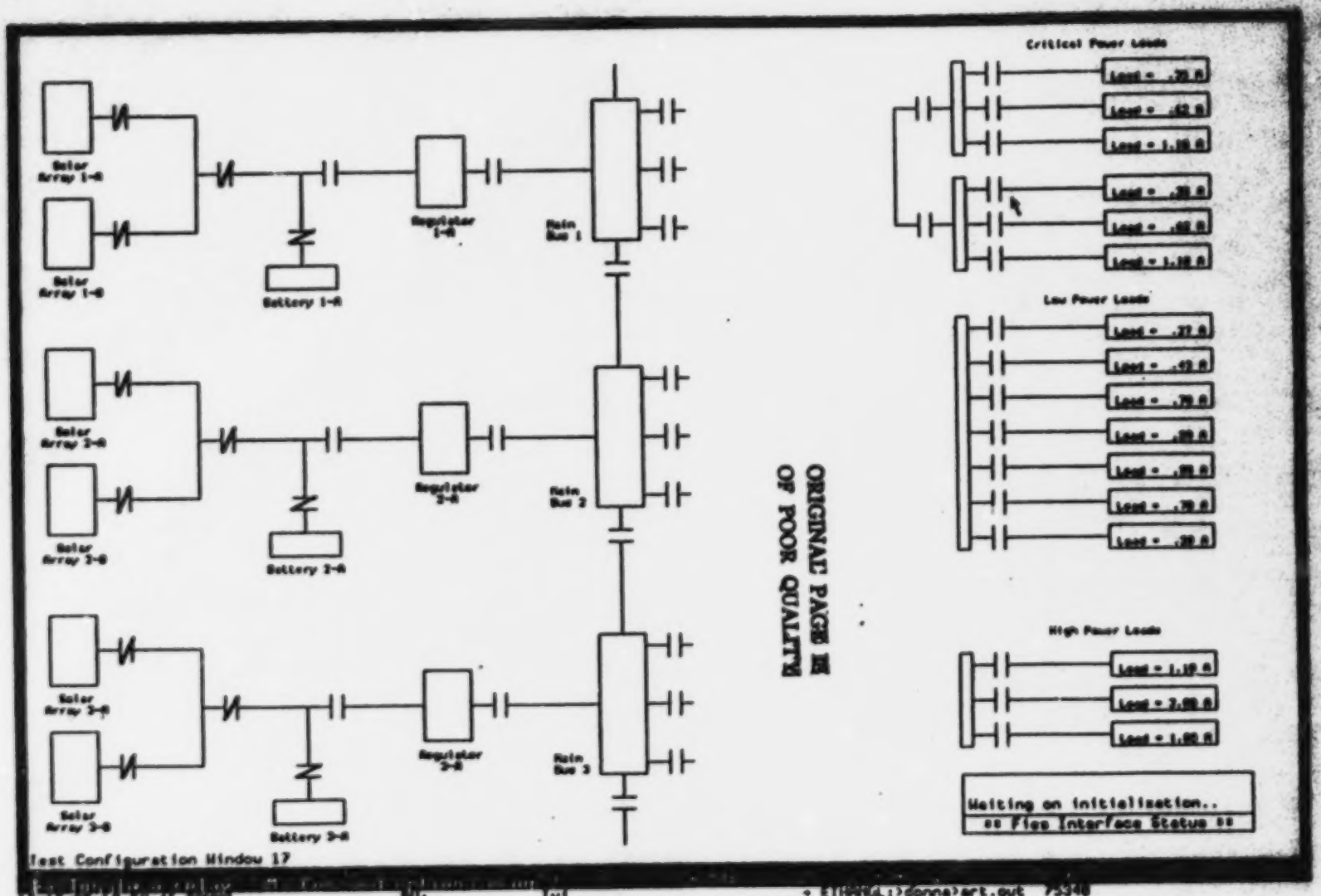
10/15/85 13:59:46 DONNA NU: Tyl * ETHANOL:>donna>art.out 75348

Figure 6-1



ORIGINAL PAGE IS OF POOR QUALITY

Figure 6-2



ORIGINAL PAGE IS
OF POOR QUALITY

Figure 6-3

6.4 Breadboard Readiness Checkout Procedure

6.4.1 Because the microcomputer system "learns" what the correct voltage and current conditions are for a given configuration during the download phase, it is important to be sure that the what is "learned" is proper for the configuration.

Some examples of improper downloading are:

a) A fault exists in the network from a previous run. If the fault was not removed prior to downloading a new configuration, this fault is learned as being part of the new configuration.

b) Power Supply voltages not set properly. If one of the desired power supplies has been turned off or has had its voltage re-adjusted for battery charging, the over/under voltage condition will be treated as proper.

c) Batteries not fully charged. If one of the batteries has had significant current used, power will be diverted from the intended path to charge the battery. This can result in insufficient power being available to drive the network, or cause the battery voltage to drift during a run.

d) Downloading of configurations which cause the breadboard protection circuitry to activate.

6.4.2 To insure that the downloaded configuration will be meaningful, two important steps must be followed.

6.4.2.1 Insure that the desired configuration is both meaningful and allowable from an electrical sense.

The network is fairly versatile and will allow improper configurations to be placed upon it. Those conditions which are likely to damage the network will be removed, but the configuration imposed on the network will undoubtedly be meaningless.

Some examples of configurations to avoid are:

a) Those which connect the output of more than 1 voltage regulator together.

b) Those which require more power to be supplied than is available.

c) Those which charge a battery from a module other than the one in which the battery resides.

d) Those which charge a deeply depleted battery.

A tool has been made available to assist in "trying out" selected configurations prior to linking the configuration with expert system. This tool is interactive and allows the user to verify that the configuration is meaningful. This tool is described in Chapter 8, titled Test Configuration Utility.

ACTION	VERIFICATION OF COMPLETENESS
Place rotary switch in `Reset' position.	Message "System has been reset" will be visible in the alphanumeric display.
Place all toggle switches on the fault insertion panel to the center or `normal' position.	No LED's will be on, and no current will be drawn from any of the H.P. supplies.
If current is still being drawn from the H.P. supplies, turn supply off, the on. This resets the SCR's which provide the direct short faults.	No current is being drawn from the H.P. supplies.
Reset the circuit breakers on the battery modules by pressing the center pole of the breaker in.	The breaker center pole is plush with the breaker "plate".
Verify that the batteries have a normal voltage. Place the rotary switch in the `status' position. Key in 22 on the thumbwheel switch. depress the `Repeat Function' button. Repeat for 23, 24.	The alphanumeric display contains the normal unloaded voltage for a fully charged battery, 33.75V.
Place the rotary switch in the `slave' position.	Message `SYMBOLICS SRC MODE' will be visible in the alphanumeric display

6.4.2.2 Secondly, the breadboard must be ready to accept the meaningful configuration. Following the steps below will place the breadboard in a condition for receiving configurations.

6.4.3 The breadboard is now ready to accept a new configuration from the Symbolics machine. After the configuration has been downloaded, and the message 'STEADY STATE ACHIEVED' is visible, the breadboard is ready to be faulted.

As an additional verification that the downloaded configuration is proper check that the front panel lamps (corresponding to the activated load areas) are glowing with full intensity. This indicates that the network has received full power to the loads.

CHAPTER 7

BATTERY MAINTENANCE AND RECHARGING

BATTERY MAINTENANCE AND RECHARGING

7.0 OVERVIEW

The Power Subsystem Automation Study Breadboard incorporates three Nickel-Cadmium batteries. These batteries are composed of 25 cells, mounted in a stainless steel case. The cells have a capacity of approximately 14 amp-hours. The batteries were manufactured in Colorado Springs, Colorado by Eagle-Picher Industries. Connection to the batteries is made through connector type MS3124T-10-6P. Pins A, B, C, are the positive terminal of the battery, while pins D, E, F are the negative terminal.

The batteries are able to be trickle charged by the breadboard. This is primarily intended to be a "station keeping" function rather than a complete charge ramp up. The relay connecting the battery to the power network does not need to be closed, nor does the circuit breaker need to be reset. Extra power available from the Hewlett-Packard power supplies is diverted to replenishing the consumed power. This allows configurations to be tested on the breadboard which don't actually use the battery as a power source.

7.1 * * * CAUTION * * *

If a configuration was placed on the breadboard which required the battery to supply a large portion of its capacity, the resulting trickle charge occurring on subsequent runs may cause false error reporting to the Expert System. The errors can result from the following situation:

1) Some configuration was run in which a large amount of current was drawn from a battery. The run completed normally, leaving the battery in a state which will require more than 10% of the H.P. supplies capacity to re-charge.

2) A new configuration is downloaded to the breadboard for testing. This configuration does not use the battery, but does connect the power supplies of the same power module to the network. Trickle charging of the battery begins.

3) The microprocessor "learns" the new configuration voltages and currents and ships them to the expert system. The microprocessor issues the message "Steady State Achieved" and awaits a fault.

4) During this time the battery is re-charging. Its voltage is increasing, and the trickle charge current is diminishing. The microcomputer will determine that this is a fault, and erroneously uplink fault data.

This situation can be avoided if a charge situation is placed on the battery sufficient to reduce the trickle current to 200 mA or less prior to the subsequent runs.

7.2 Periodic Maintenance

The batteries will also require periodic maintenance to assure continued stable characteristics. This involves performing a deep cycle discharge and recharging. The procedure outlined in the Battery Maintenance Manual (supplied by Eagle-Picher) should be used. This procedure may be followed by removing the batteries to a remote location where a constant current source is available. Alternatively, the procedure may be closely approximated by leaving the batteries in the system and following the steps outlined below:

7.2.1 Simultaneous discharge of batteries

The batteries may be discharged simultaneously while in the breadboard at a one hour rate. This is useful in observing the characteristics of the batteries. Loads equivalent to 1.1 amperes can be obtained on each of the load busses in the network. A battery in each power module can be connected to these equivalent 1.1 ampere loads.

Figure 7.1 shows the discharge characteristic for the battery in module 3. Notice that the battery drops quickly from its unloaded voltage of approximately 35 volts to the normal loaded voltage of 31.6 volts. From here the voltage remains relatively stable until the dropoff voltage of 29.3 volts. Here the battery output avalanches to 20 volts within a period of several minutes.

As the voltage regulation scheme chosen for the breadboard is linear, and requires approximately 2.0 volts of margin between the unregulated source and its output, the battery actually has a useful lifespan of 13 AMP-HOURS.

7.2.2 Discharge of a single battery

A single battery may be rapidly discharged at up to a 5 A-H rate by connecting loads on each of the high power, low power, and critical busses.

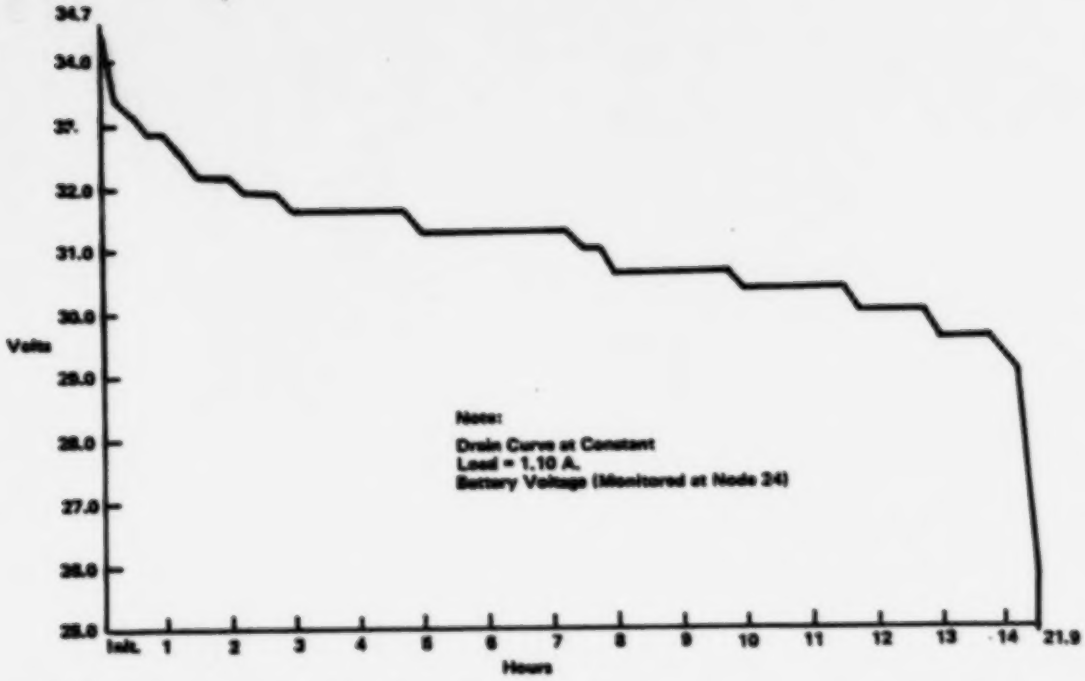


Figure 11 PSAS Battery No. 3 Discharge Characteristics

7.2.3 Charging of Batteries

The breadboard is able to charge the batteries in a "constant potential" mode only. However, a constant current charge can be approximated. This approach requires operator intervention at intervals of 1 hour, but has the advantage of completing the charge cycle in less than 1 day. Also, all three batteries may be charged in the system at the same time. Follow the steps below to charge the battery from a discharged condition:

1) Initialize the breadboard and the microcomputer system. Place the network in a configuration where the relays are all open.

2) Using a voltmeter connected to the sense jacks on the H.P. supply faceplate, set the voltage on both power channels from 38.5 volts to 40.5 volts. Assure that the current limit on the supply is wide open.

3) Close the relays along the path from the supplies to the battery, leaving all other relays open. Verify that current is being drawn evenly from both power supplies by observing the meters on the H.P. supplies. Adjust the voltage until a balance of current is drawn from both power channels in the 6255A supply.

4) Using the front panel display; examine the battery voltage. This is available at nodes 22, 23, 24 for the batteries in power modules 1, 2, and 3 respectively. Notive that the current sensors at the battery are disabled and forced to zero. The initial battery voltage should be approximately 33 volts.

5) Monitor the current flowing into the batteries in modules 1, 2, 3 at the nodes 3, 9, 15. The initial reading of battery current at this point should be approximately - 3.0 Amperes. However, this reading will rapidly diminish to the - 2.5 A range. The minus sign at this node indicates that current is flowing into the battery.

6) Continue monitoring the current flowing into the battery. Usually, once an hour will be sufficient to re-adjust the current. As the current drops below 2.0 Amperes, adjust the voltage upwards on both power channels of the H.P. supply. As a guideline, increasing the voltage on each power channel by .5V will return the battery current to the 2.5A range.

7) Figure 7.2 is an example of this type of charging practice used on the battery in power module 3. The charging cycle may be terminated when the accumulated charge (the area under the current plot) reaches 14 AMP-HOURS. Alternatively, the charge may be terminated when the battery voltage displays the "head and shoulders" characteristic shown in the figure.

The battery protection hardware will prevent the battery from overcharging (by shunting away excess current and limiting the raise in battery voltage to 39.0V), but charging should be terminated when either of the above two conditions is reached.

* * * CAUTION * * *

Finally, return the H.P. supply voltages to 38.5 volts. Failure to do this will overdrive the voltage regulator when power is applied to the regulator circuit from the supplies.

The battery connections to the battery protection circintry have been sealed in plastic cement. Do not attempt to defeat this protection. Never remove the batteries from the battery module unless they have been drained of power first.

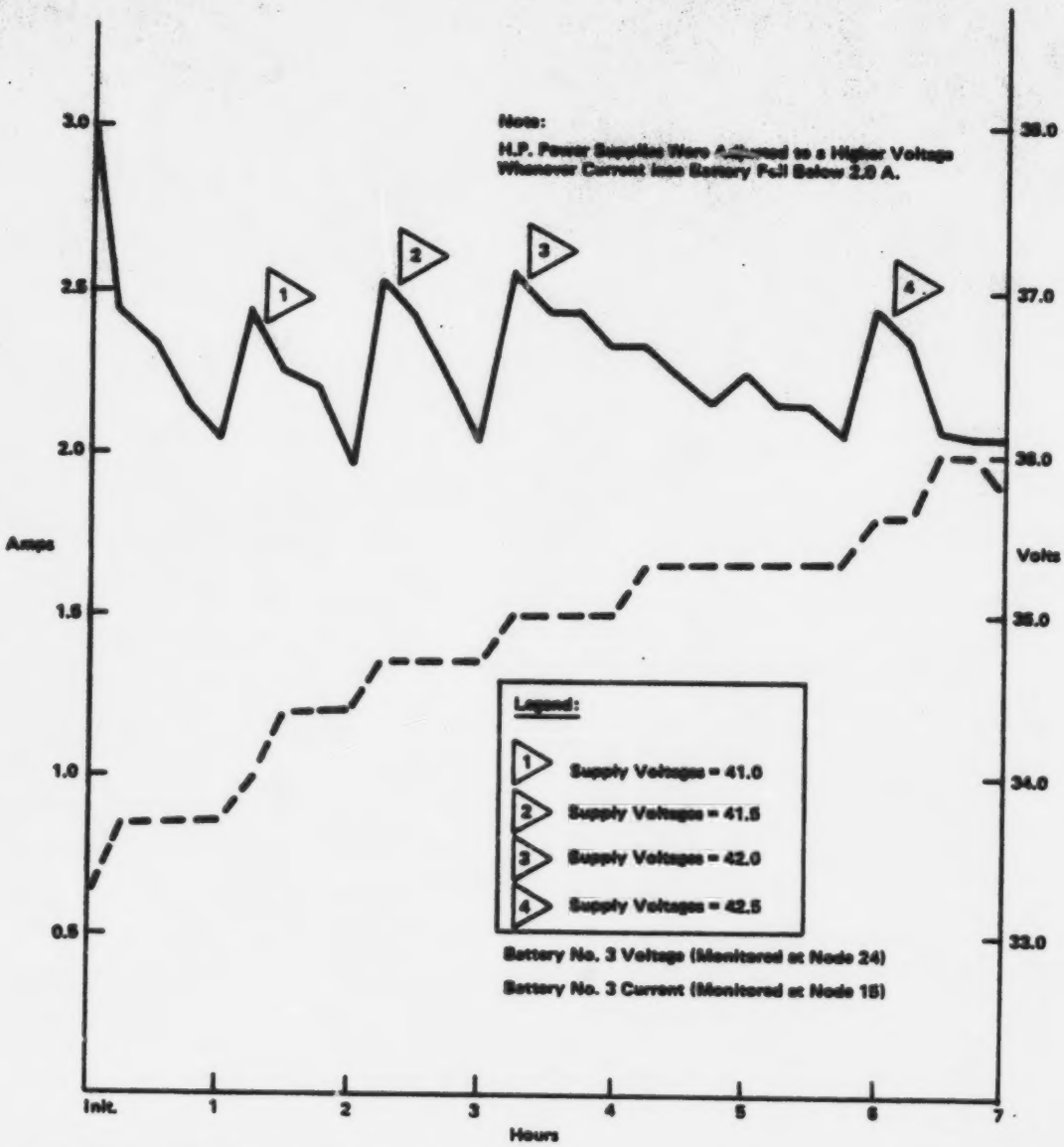


Figure 2 PSAS Battery No. 3 Charge Characteristics

CHAPTER 8

TEST CONFIGURATION UTILITY

TEST CONFIGURATION UTILITY

8.0 Overview

This section provides a description of the test configuration utility. The purpose of this program is to allow the user to exercise the interface between the microprocessor and the LISP machine without the overhead involved in a full ART run. This program will be used for system verification prior to a run or for problem determination.

8.1 Program Startup

The program is invoked from a LISP LISTENER by entering (test-configuration).

8.2 Test Configuration Options

The following options are available through the menu system. The main menu is handled exactly as described in chapter 4, the only difference being an expanded set of command options.

8.2.1 Download Configuration

This option is used to download a configuration to the microprocessor prior to a run. The configuration is specified either by using the RETRIEVE CONFIGURATION menu option to load a previously saved configuration into the display or by clicking the required relays off or on using the mouse.

8.2.2 Refresh

This option causes the microprocessor to transmit the current state of the breadboard to the LISP machine. The data can then be viewed through the EXAMINE BREADBOARD DATA option.

8.2.3 Retrieve Configuration

This option is used to load a previously saved configuration into the display from the fies-library-directory. Upon clicking left on this menu item the user should observe a window at the top of the screen. The window may be exited by using the ABORT key.

Once the window is visible, type in the name of the file corresponding to the desired configuration. After the name is entered, hit ‹RETURN›. The window should disappear and the user should observe that the configuration has been loaded into the display. If the file does not exist the message "file does not exist" will appear in the FIES Interface Status window.

8.2.4 Archive Configuration

This option is used to store a configuration that is currently loaded in the display into a file in the fies-library directory.

After selecting this option, the user should observe a window at the top of the display. Enter the name of the file in which the configuration is to be stored and hit ‹RETURN›. If the file already exists the message "File exists" will appear in the FIES Interface status window. In this case, the user will be reprompted for a new file name. Enter the new file name or use the ABORT key to exit the window.

While any naming conventions may be used, it is suggested that a common extension such as ".cnf" be employed. This is the extension use for MMC provided configuration files. A directory name need not be entered, as the default will be to store the files in c: › fies › fies-library.

8.2.5 Examine Steady State Data

This option allows the user to examine the state of the breadboard following a download and the subsequent transmission of the steady-state table to the LISP machine. To return from the steady state window display, the user should click the middle mouse button and select the option "RETURN TO PREVIOUS SCREEN".

8.2.6 Examine Breadboard Data

This option may only be used in the interactive version of the system. The option allows the user to examine the state of the breadboard following fault insertion and the subsequent transmission of the fault table to the LISP machine. To return from the breadboard window display, the user should click the middle mouse button and select the option "RETURN TO PREVIOUS SCREEN".

8.2.7 Retrieve Scenario

This option allows the user to retrieve a data set that has archived a breadboard configuration, the corresponding steady state and a faulted configuration. This data may be used to verify current breadboard values or to drive an ART run.

8.2.8 Archive Scenario

This option allows a user to save a configuration, the corresponding steady state table and a faulted data set. This is used to save data sets for breadboard verification or for testing independent of the breadboard.

8.2.9 Art Command Window

This option passes control to ART. Typically, this option will be used following a Retrieve Scenario command, where the intent is to exercise the ART application independently of the breadboard.

8.2.10 QUIT

This option causes a return to the LISP LISTENER.

APPENDICES

APPENDIX A

Operation Flowcharts and Screens

TUTORIAL		TEXT
Overview	Hardware Initialization	Section 2.1
	Software Initialization	Section 2.2
	Run Time	Section 2.3

TUTORIAL		TEXT	SCREENS
Hardware Initialization	Power Up Breadboard	2.1	
	Remove Faults from Breadboard	2.2	
	Verify Completion of Microprocessor Self Test	2.3	
	Boot Symbolics Processor	2.4	2.1.4

Symbolics 3670™ System

This machine is *Martin Marietta Denver Aerospace - Machine Intelligence Lab Einbecker*

Symbolics™ System, Release 6.1
Loaded from FEP0:>mil-r6-1-color.load.1
1536K words Physical memory, 37500K words Swapping space.

Release	6.1
COLOR	135.50
COLOR-SEND	68.7
IMAGES	56.21
FEP	127

You are typing to *Lisp Listener 1*. Control characters are interpreted as commands to edit input. Type Control-<HELP> for a list of input editor commands.

Use the ":Help" command to display a list of all the Command Processor commands.
Type <SELECT> D to select the Document Examiner to read online documentation.
Type <SELECT> <HELP> for a list of programs.
Type <SELECT> <HELP> for a list of asynchronous and window operations.
Click the rightmost mouse button to select the System Menu of programs and window operations.
Type Symbol-<HELP> for a list of special function keys and special character keys.

Please login.
■

(c) Copyright 1985, 1984, 1983, 1982, 1981, 1980, Symbolics, Inc.
All Rights Reserved.

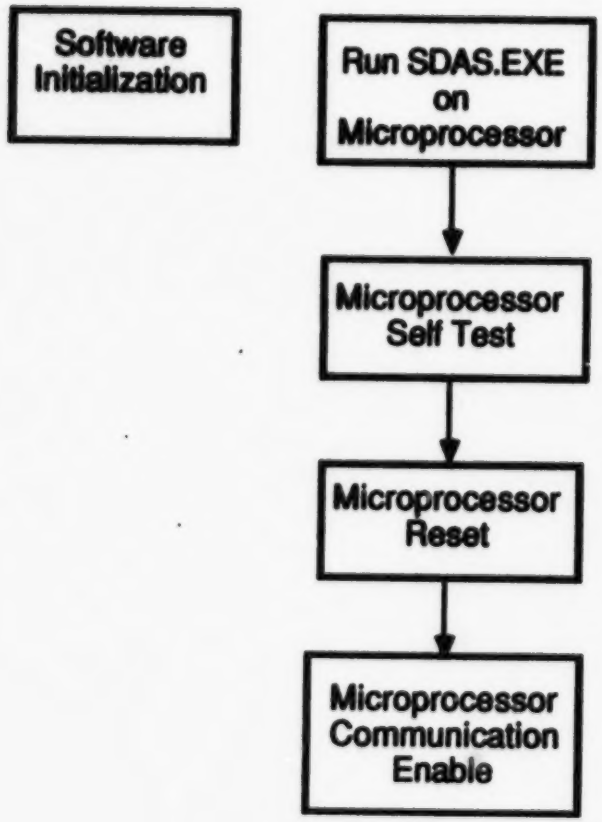
Use the command "Show Legal Notice" to see important legal notices.

Symbolics, Symbolics 3600 and Symbolics 3670 are trademarks of Symbolics, Inc.

Lisp Listener 1

SCREEN 2.1.4

TUTORIAL TEXT SCREENS



2.2.1

2.2.2

2.2.3

2.2.4

TUTORIAL		TEXT	SCREENS
Software Initialization	Login to Symbolics	2.2.5	2.2.5
	Select ART	2.2.6	
	Clear ART	2.2.7	2.2.7
	Load FIES Application	2.2.8	2.2.8.1, 2.2.8.2
	Reset ART	2.2.9	2.2.9

```
Loading DIOXIN:>fies-art-production>fies-xmit-relay-configuration.bin
Loading DIOXIN:>fies-art-production>fies-set-art-sensors.bin
Loading DIOXIN:>fies-art-production>fies-set-art-fault-sensors.bin
Loading DIOXIN:>fies-art-production>fies-power-distribution-lines-funcs.bin
Loading DIOXIN:>fies-art-production>fies-undraw-power-distribution-lines.bin
Loading DIOXIN:>fies-art-production>fies-redraw-power-distribution-lines.bin
Loading DIOXIN:>fies-art-production>fies-reset-power-distribution-lines.bin
Loading DIOXIN:>fies-art-production>fies-archive-configuration.bin
Loading DIOXIN:>fies-art-production>fies-retrieve-configuration.bin
Loading DIOXIN:>fies-art-production>fies-archive-scenario.bin
Loading DIOXIN:>fies-art-production>fies-retrieve-scenario.bin
Loading DIOXIN:>fies-art-production>fies-xmit-request.bin
Loading DIOXIN:>fies-art-production>fies-parse-raw-data-for-scenario.bin
Loading DIOXIN:>fies-art-production>fies-start-art-scenario.bin
Loading DIOXIN:>fies-art-production>fies-send-lisp.bin
Loading DIOXIN:>fies-art-production>fies-initialize-mouse.bin
Loading DIOXIN:>fies-art-production>fies-initial-configuration.bin
Loading DIOXIN:>fies-art-production>fies-serial-interface.bin
Loading DIOXIN:>fies-art-production>fies-test-configuration.bin
Loading DIOXIN:>fies-art-production>fies-serial-interface-test.bin
Loading DIOXIN:>fies-art-production>fies-refresh-data-display.bin
Loading DIOXIN:>fies-art-production>fies-serial-interface-level-2.bin
Loading DIOXIN:>fies-art-production>fies-steady-state-data-display.bin
Loading DIOXIN:>fies-art-production>fies-fault-data-display.bin
Loading DIOXIN:>fies-art-production>fies-display-faults-from-art.bin
T
```

You are typing to *Lisp Listener 1*. Control characters are interpreted as commands to edit input. Type Control-[HELP] for a list of input editor commands.

Use the ":Help" command to display a list of all the Command Processor commands.
Type [SELECT] D to select the Document Examiner to read online documentation.
Type [SELECT] [HELP] for a list of programs.
Type [SHIFT-SELECT] [HELP] for a list of asynchronous and window operations.
Click the rightmost mouse button to select the System Menu of programs and window operations.
Type Symbol-[HELP] for a list of special function keys and special character keys.

```
(login 'fies-art-production)
Loading DIOXIN:>fies-art-production>lispm-init.lisp into package USER
Loading DIOXIN:>fies-art-production>make-fies-interface.lisp into package ART-USER
Loading DIOXIN:>fies-art-production>fies-interface-defs.bin
Loading DIOXIN:>fies-art-production>fies-get-edges.bin
Loading DIOXIN:>fies-art-production>fies-clear-status-window-display.bin
Loading DIOXIN:>fies-art-production>fies-display-status-message.bin
Loading DIOXIN:>fies-art-production>fies-get-configuration-file-name.bin
Loading DIOXIN:>fies-art-production>fies-make-configuration-file-name.bin
Loading DIOXIN:>fies-art-production>fies-get-scenario-file-name.bin
Loading DIOXIN:>fies-art-production>fies-make-scenario-file-name.bin
Loading DIOXIN:>fies-art-production>fies-undraw-selected-configuration.bin
Loading DIOXIN:>fies-art-production>fies-draw-selected-configuration.bin
Loading DIOXIN:>fies-art-production>fies-set-art-relays.bin
```

Lisp Listener 1

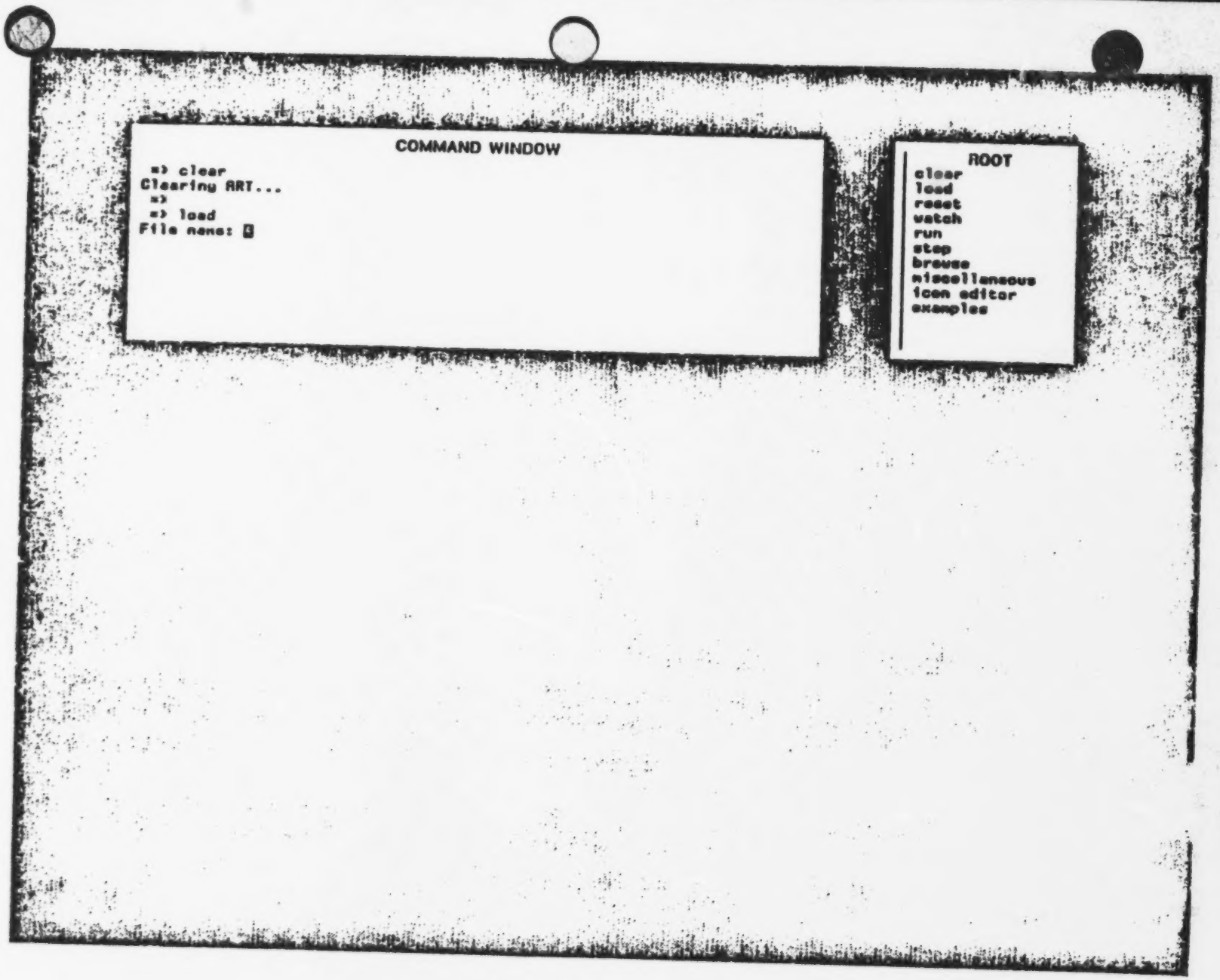
SCREEN 2.2.5

ORIGINAL PAGE IS
OF POOR QUALITY

```
COMMAND WINDOW

=> clear
Clearing ART...
=>
=>
```

```
ROOT
clear
load
reset
watch
run
step
browse
miscellaneous
icon editor
examples
```

SCREEN 2.2.7


COMMAND WINDOW
=> clear
Clearing ART...
=>
=> load
File name:
ROOT
clear
load
reset
watch
run
step
browse
miscellaneous
icon editor
examples

SCREEN 2.2.8.1

```
COMMAND WINDOW
Compiling rule SENSOR-INCONSISTENCY-CHECK-SX... =P=P=J=P=J=P=J=P=J=P=J
=P=J=P+J=P+J=P+J
Loading DIGXIN:>art-test>p-summary.art.1 in package ART-USER and base
10.
Compiling rule PRINTOUT-ALL-FAULTS... =P=J
Compiling rule SUMMARIZE-ALL-FAULTS... =P=P+J
Compiling rule PRINTOUT-REMAINING-FAULTS-FLAG... =P=J
Compiling rule PRINTOUT-REMAINING-FAULTS... =P=P=J=P+J+P+J
 => [redacted]
=P+J=P+J=P+J=P+J=P+J
Compiling rule SENSOR-INCONSISTENCY-CHECK-SOURCE... =P=P=J=P=J=P=J=P+J
=P+J=P+J=P+J=P+J=P+J
```

```
ROOT
clear
load
reset
watch
run
step
browse
miscellaneous
icon editor
examples
```

ORIGINAL PAGE IS
OF POOR QUALITY

SCREEN 2.2.8.2

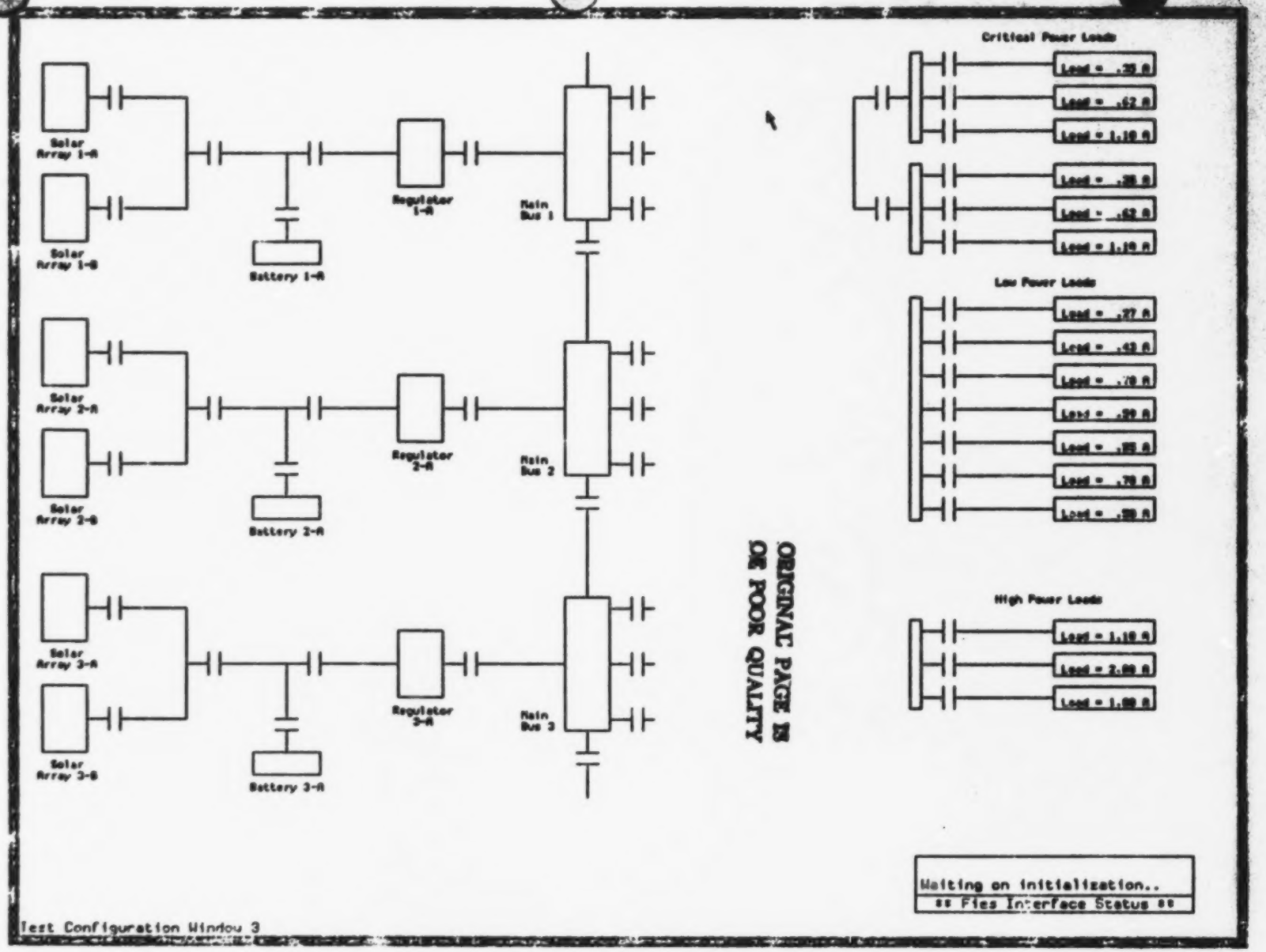
```
COMMAND WINDOW
Compiling rule SENSOR-INCONSISTENCY-CHECK-SK... =P=P=J=P=J=P=J=P=J=P=J
=P=J=P+J=P+J=P+J
Loading DIOKIN:>art-test>p-summary.art.1 in package ART-USER and base
10.
Compiling rule PRINTOUT-ALL-FAULTS... =P=J
Compiling rule SUMMARIZE-ALL-FAULTS... =P=P+J
Compiling rule PRINTOUT-REMAINING-FAULTS-FLAG... =P=J
Compiling rule PRINTOUT-REMAINING-FAULTS... =P=P=J=P+J+P+J
 => reset
Resetting ART...
 => 
=P+J=P+J=P+J=P+J=P+J
```

```
ROOT
clear
load
reset
watch
run
step
browse
miscellaneous
text editor
examples
```

SCREEN 2.2.9

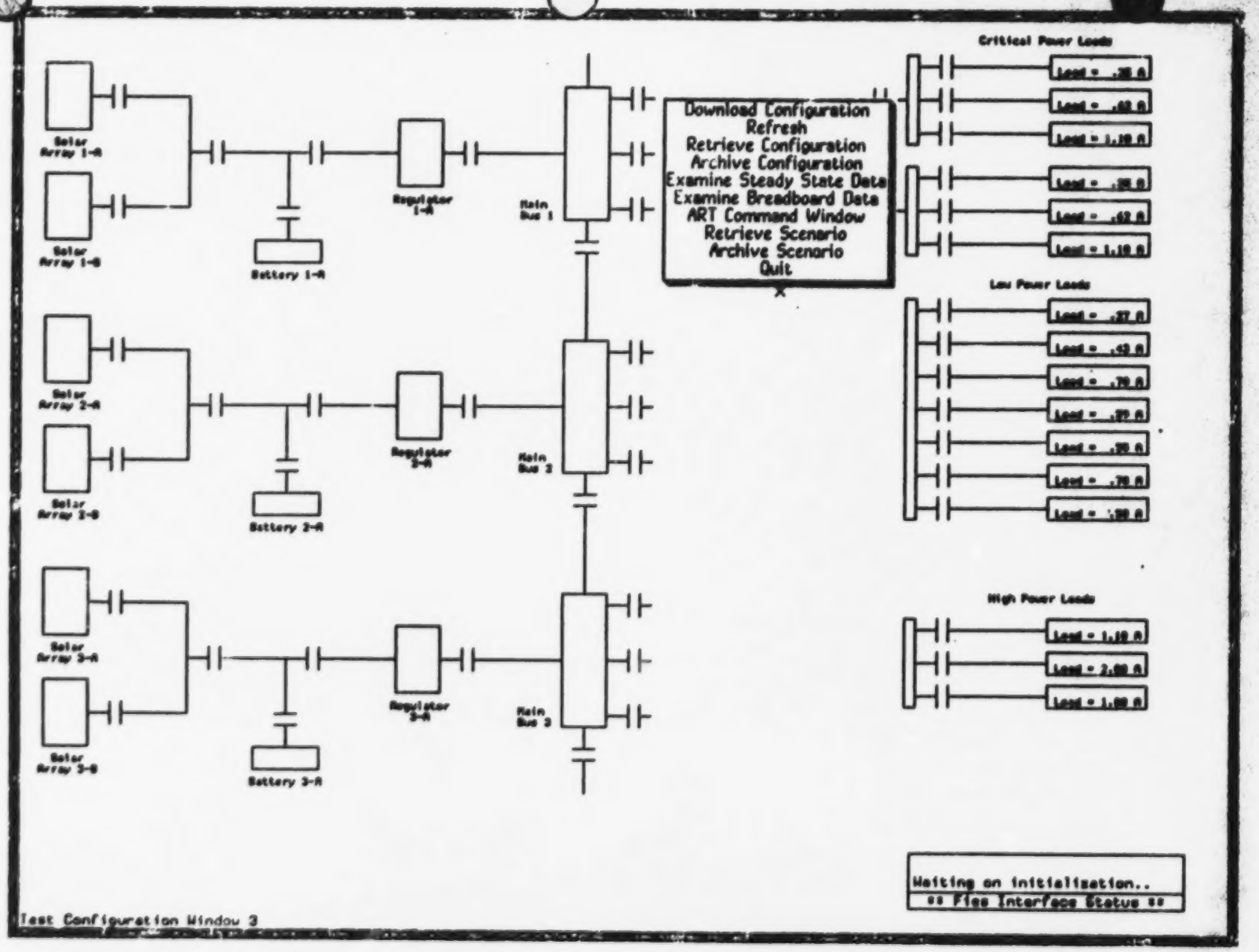
TUTORIAL

		TEXT	SCREENS
Runtime	Enter Run in ART Window	2.3.1	2.3.1
	In Symbolics Graphics Display Bring Up Main Menu	2.3.2	2.3.2
	Select Retrieve Configuration Option	2.3.3	2.3.3.1, 2.3.3.2
	Enter Configuration File Name	2.3.4	2.3.4
	Observe Configuration in Graphic Display	2.3.5	



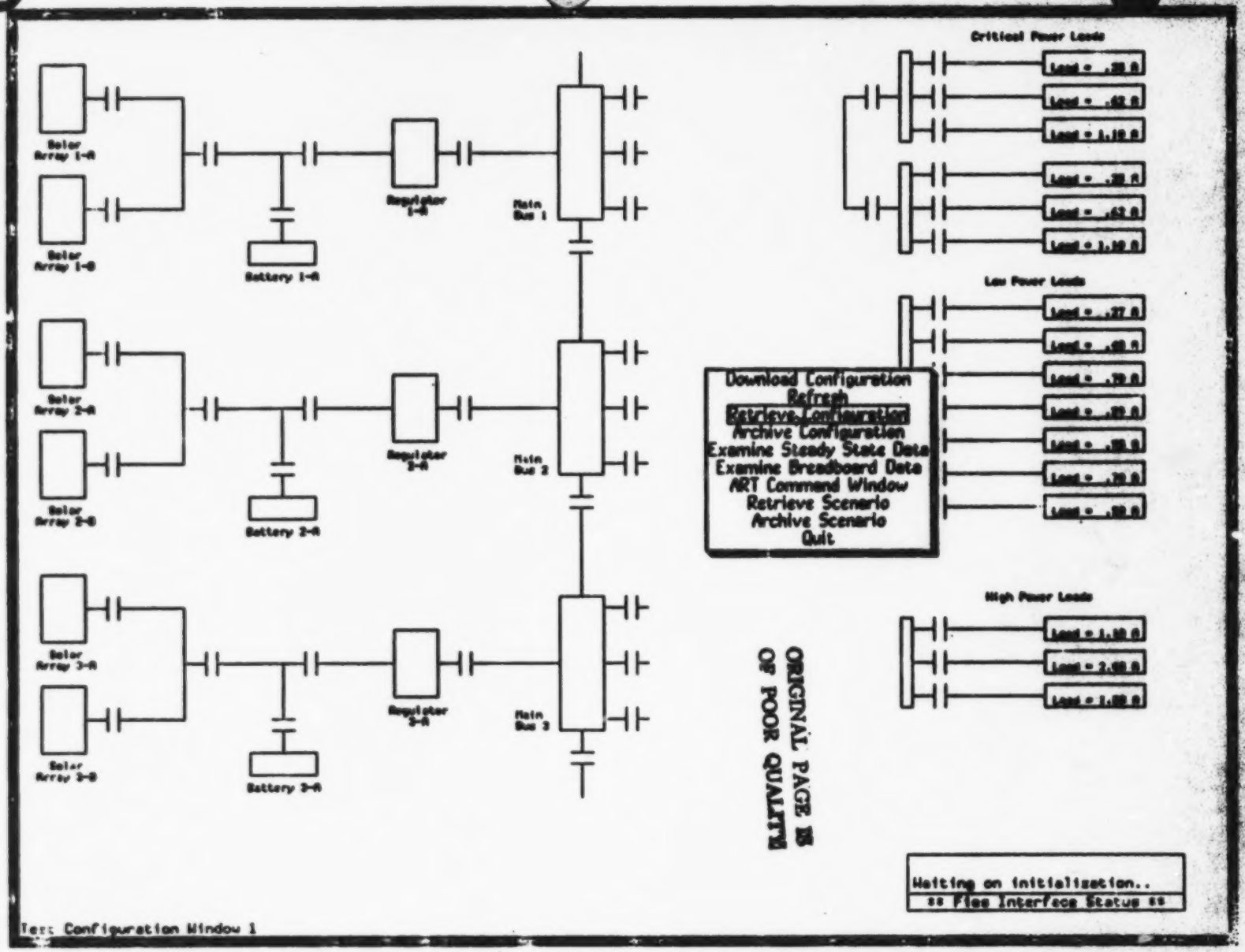
ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 2.3.1



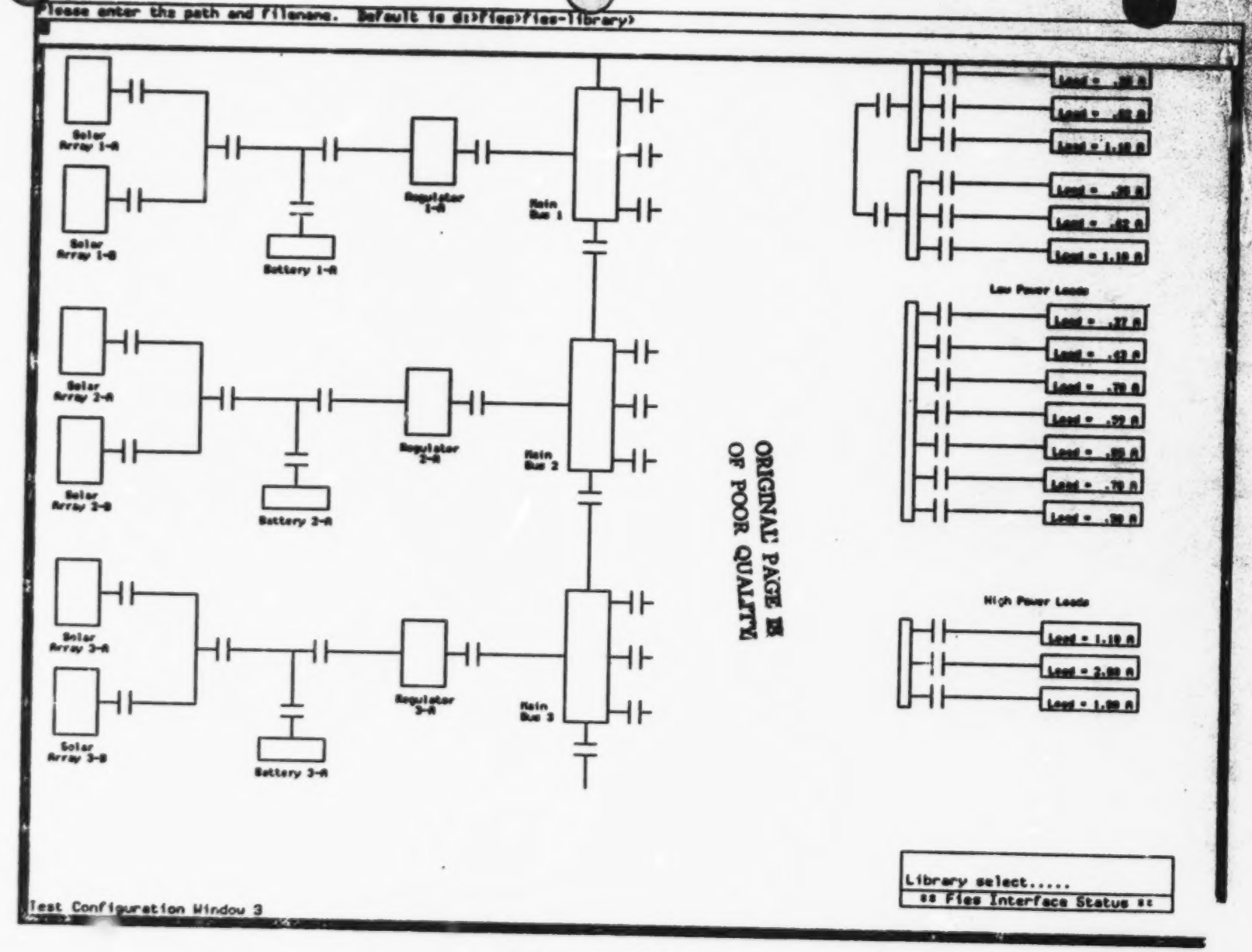
SCREEN 2.3.2

ORIGINAL PAGE IS
OF POOR QUALITY



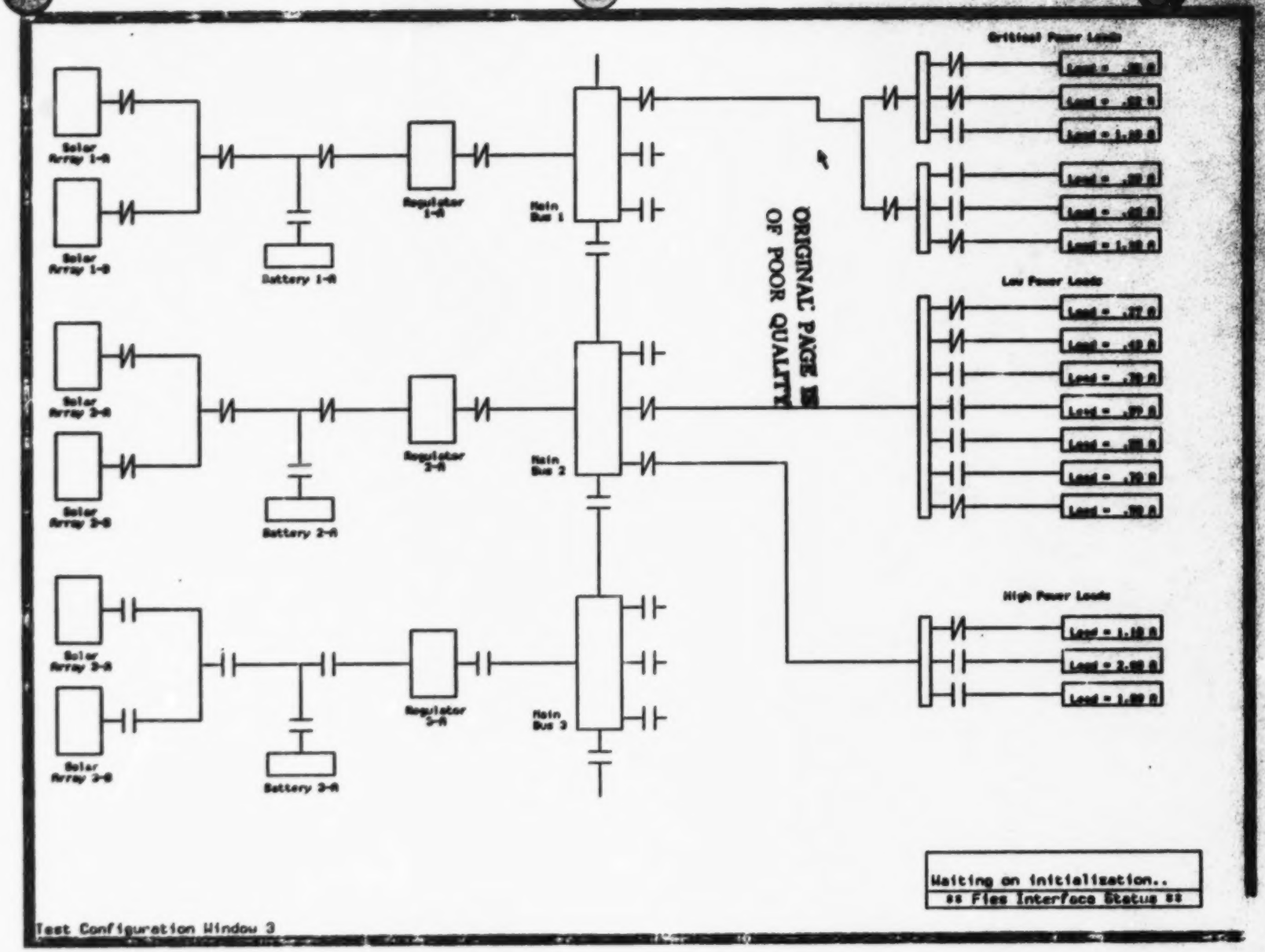
ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 2.3.3.1



ORIGINAL PAGE IS OF POOR QUALITY

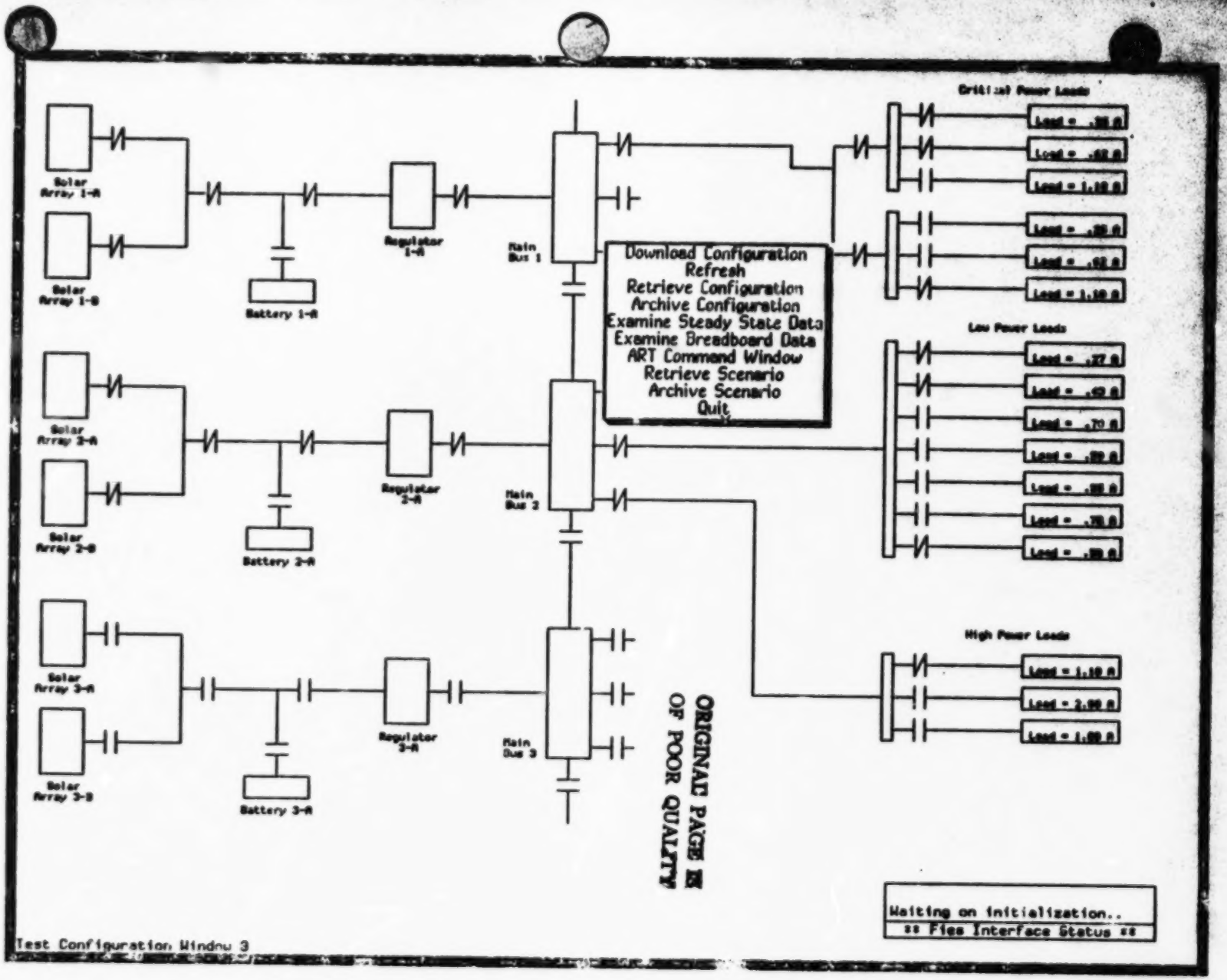
SCREEN 2.3.3.2



ORIGINAL PAGE IS
OF POOR QUALITY

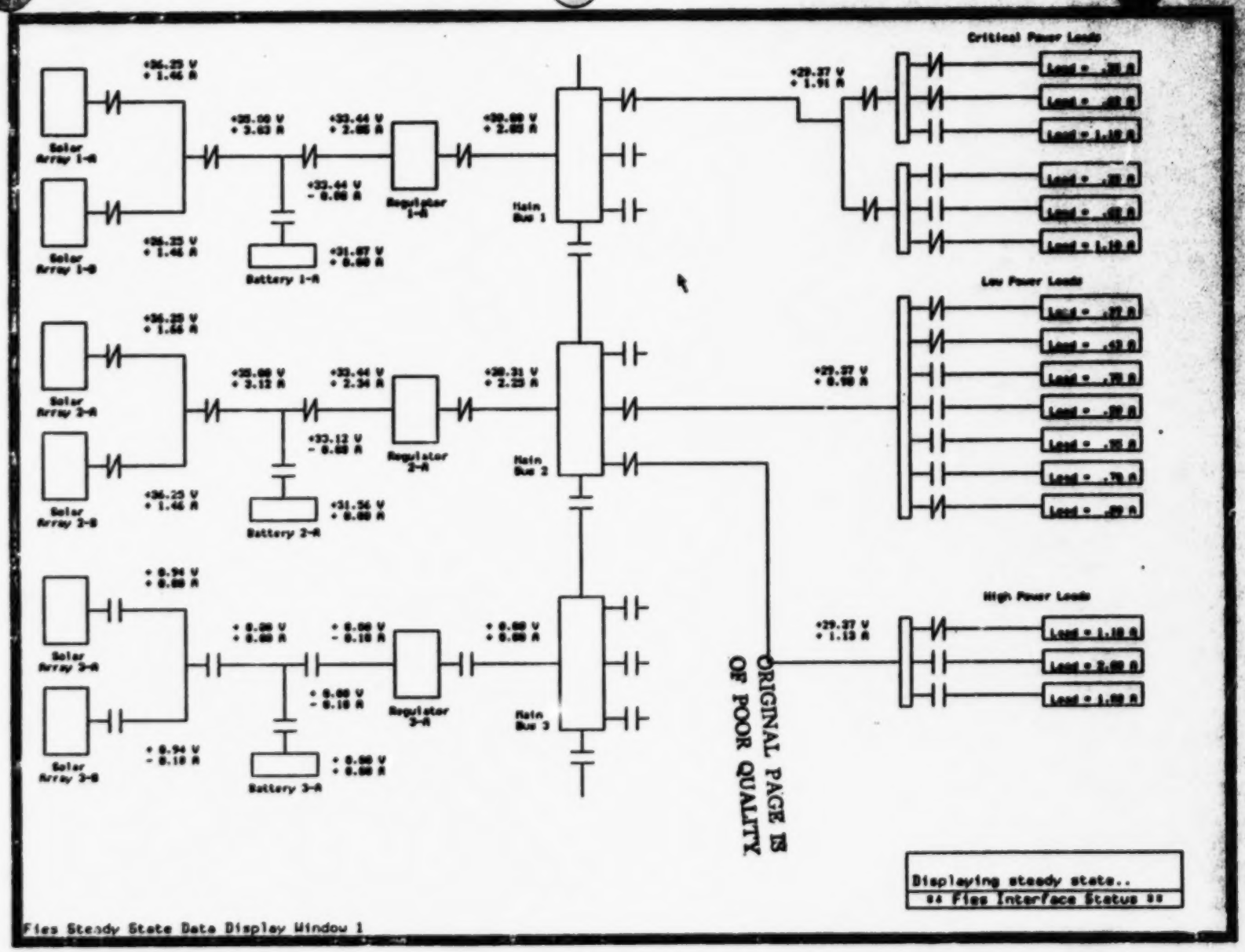
SCREEN 2.3.4

TUTORIAL		TEXT	SCREENS
Runtime	BRING UP MAIN MENU AGAIN	2.3.6	2.3.6.1
	SELECT DOWNLOAD CONFIGURATION OPTION	2.3.7	2.3.6.2
	OBSERVE "I/O WAIT" FAULT DATA MESSAGE ON SYMBOLICS	2.3.8	
	INSERT FAULT IN BREADBOARD	2.3.9	
	WAIT FOR AND VERIFY RESULTS	2.3.10	



ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 2.3.6.1



ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 2.3.6.2

PRODUCTION		TEXT
Overview	Hardware Initialization	3.1.1
	Software Initialization	3.1.2
	Run Time	3.1.3

PRODUCTION

		TEXT	SCREENS
Hardware Initialization	Power UP Breadboard	3.1.1.1	
	Remove Faults From Breadboard	3.1.1.2	
	Verify Completion of Microprocessor Self Test	3.1.1.3	
	Boot Symbolics Processor	3.1.1.4	3.1.1.4

Symbolics 3670™ System

This machine is ***Martin Marietta Denver Aerospace - Machine Intelligence Lab Einbecker***

Symbolics™ System, Release 6.1
Loaded from FEP0:>mil-r6-1-color.load.1
1536K words Physical memory, 37500K words Swapping space.

Release	6.1
COLOR	135.30
COLOR-DEMO	60.7
IMAGES	56.21
FEP	127

You are typing to *Lisp Listener 1*. Control characters are
interpreted as commands to edit input. Type Control-HELP
for a list of input editor commands.

Use the ":Help" command to display a list of all the Command Processor commands.
Type SYMBOL D to select the Document Examiner to read online documentation.
Type SYMBOL HELP for a list of programs.
Type COMPLETE HELP for a list of asynchronous and window operations.
Click the rightmost mouse button to select the System Menu of programs and window operations.
Type Symbol-HELP for a list of special function keys and special character keys.

Please login.
▮

(c) Copyright 1985, 1984, 1983, 1982, 1981, 1980, Symbolics, Inc.
All Rights Reserved.

Use the command "Show Legal Notice" to see important legal notices.

Symbolics, Symbolics 3600 and Symbolics 3670 are trademarks of Symbolics, Inc.

Lisp Listener 1

SCREEN 3.1.1.4

PRODUCTION		TEXT
Software Initialization	Run SDAS.EXE on Microprocessor	3.1.2.1
	Microprocessor Self Test	3.1.2.2
	Microprocessor Reset	3.1.2.3
	Microprocessor Communication Enable	3.1.2.4

PRODUCTION		TEXT	SCREENS
Software Initialization	Login to Symbolics	3.1.2.5	3.1.2.5
	Select ART	3.1.2.6	
	Clear ART	3.1.2.7	3.1.2.7
	Load FIES Application	3.1.2.8	3.1.2.8.1, 3.1.2.8.2
	Reset ART	3.1.2.9	3.1.2.9

```
Loading DIOXIN:>fies-art-production>fies-xmit-relay-configuration.bin
Loading DIOXIN:>fies-art-production>fies-set-art-sensors.bin
Loading DIOXIN:>fies-art-production>fies-set-art-fault-sensors.bin
Loading DIOXIN:>fies-art-production>fies-power-distribution-lines-funcs.bin
Loading DIOXIN:>fies-art-production>fies-undraw-power-distribution-lines.bin
Loading DIOXIN:>fies-art-production>fies-redraw-power-distribution-lines.bin
Loading DIOXIN:>fies-art-production>fies-reset-power-distribution-lines.bin
Loading DIOXIN:>fies-art-production>fies-archive-configuration.bin
Loading DIOXIN:>fies-art-production>fies-retrieve-configuration.bin
Loading DIOXIN:>fies-art-production>fies-archive-scenario.bin
Loading DIOXIN:>fies-art-production>fies-retrieve-scenario.bin
Loading DIOXIN:>fies-art-production>fies-xmit-request.bin
Loading DIOXIN:>fies-art-production>fies-parse-raw-data-for-scenario.bin
Loading DIOXIN:>fies-art-production>fies-start-art-scenario.bin
Loading DIOXIN:>fies-art-production>fies-send-lisp.bin
Loading DIOXIN:>fies-art-production>fies-initialize-mouse.bin
Loading DIOXIN:>fies-art-production>fies-initial-configuration.bin
Loading DIOXIN:>fies-art-production>fies-serial-interface.bin
Loading DIOXIN:>fies-art-production>fies-test-configuration.bin
Loading DIOXIN:>fies-art-production>fies-serial-interface-test.bin
Loading DIOXIN:>fies-art-production>fies-refresh-data-display.bin
Loading DIOXIN:>fies-art-production>fies-serial-interface-level-2.bin
Loading DIOXIN:>fies-art-production>fies-steady-state-data-display.bin
Loading DIOXIN:>fies-art-production>fies-fault-data-display.bin
T
```

You are typing to *Lisp Listener 1*. Control characters are interpreted as commands to edit input. Type Control-[illegible] for a list of input editor commands.

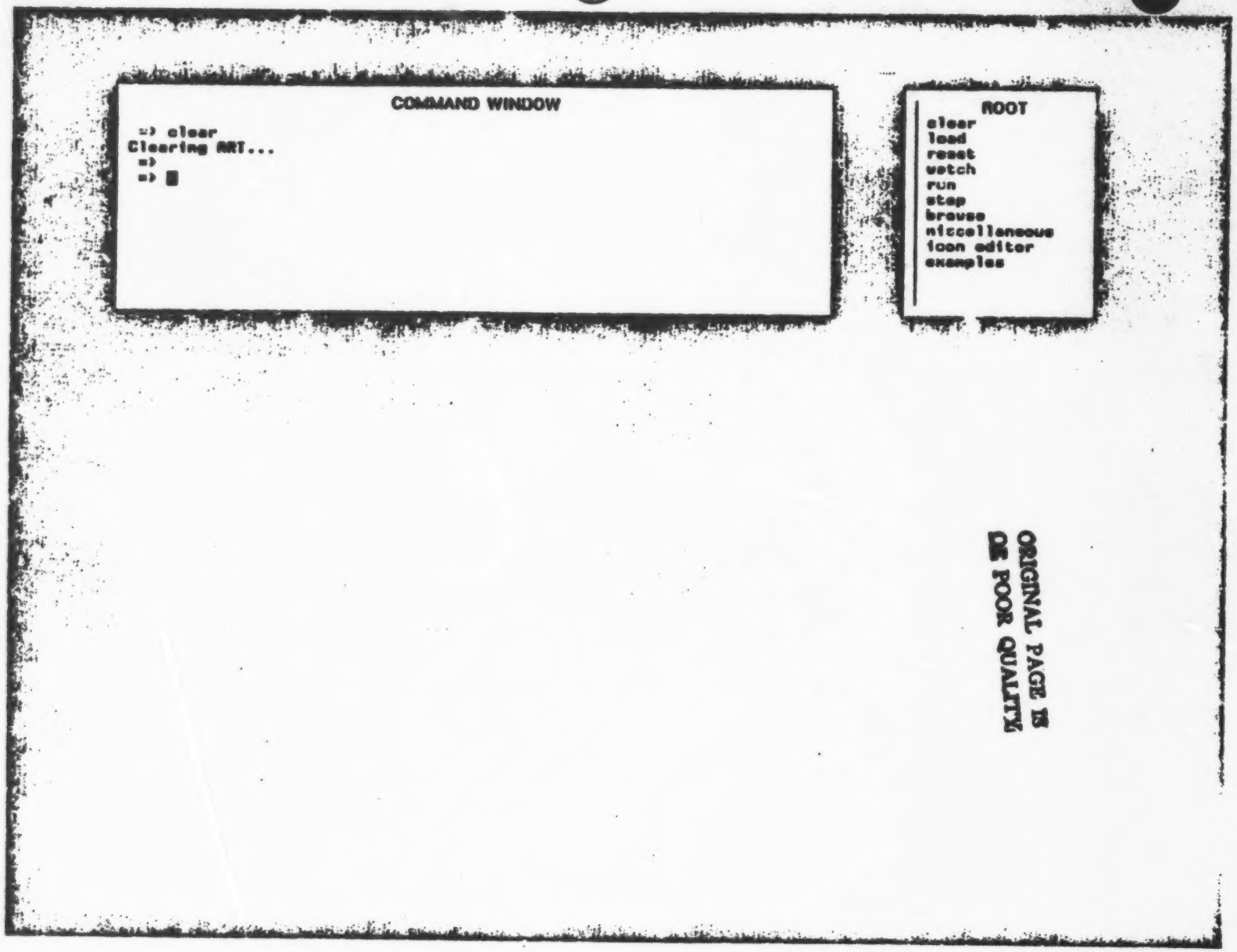
Use the ":Help" command to display a list of all the Command Processor commands.
Type [illegible] D to select the Document Examiner to read online documentation.
Type [illegible] [illegible] for a list of programs.
Type [illegible] [illegible] for a list of asynchronous and window operations.
Click the rightmost mouse button to select the System Menu of programs and window operations.
Type Symbol-[illegible] for a list of special function keys and special character keys.

```
(login 'fies-art-production)
Loading DIOXIN:>fies-art-production>lispm-init.lisp into package USER
Loading DIOXIN:>fies-art-production>make-fies-interface.lisp into package ART-USER
Loading DIOXIN:>fies-art-production>fies-interface-defs.bin
Loading DIOXIN:>fies-art-production>fies-get-edges.bin
Loading DIOXIN:>fies-art-production>fies-clear-status-window-display.bin
Loading DIOXIN:>fies-art-production>fies-display-status-message.bin
Loading DIOXIN:>fies-art-production>fies-get-configuration-file-name.bin
Loading DIOXIN:>fies-art-production>fies-make-configuration-file-name.bin
Loading DIOXIN:>fies-art-production>fies-get-scenario-file-name.bin
Loading DIOXIN:>fies-art-production>fies-make-scenario-file-name.bin
Loading DIOXIN:>fies-art-production>fies-undraw-selected-configuration.bin
Loading DIOXIN:>fies-art-production>fies-draw-selected-configuration.bin
Loading DIOXIN:>fies-art-production>fies-set-art-relays.bin
```

Lisp Listener 1

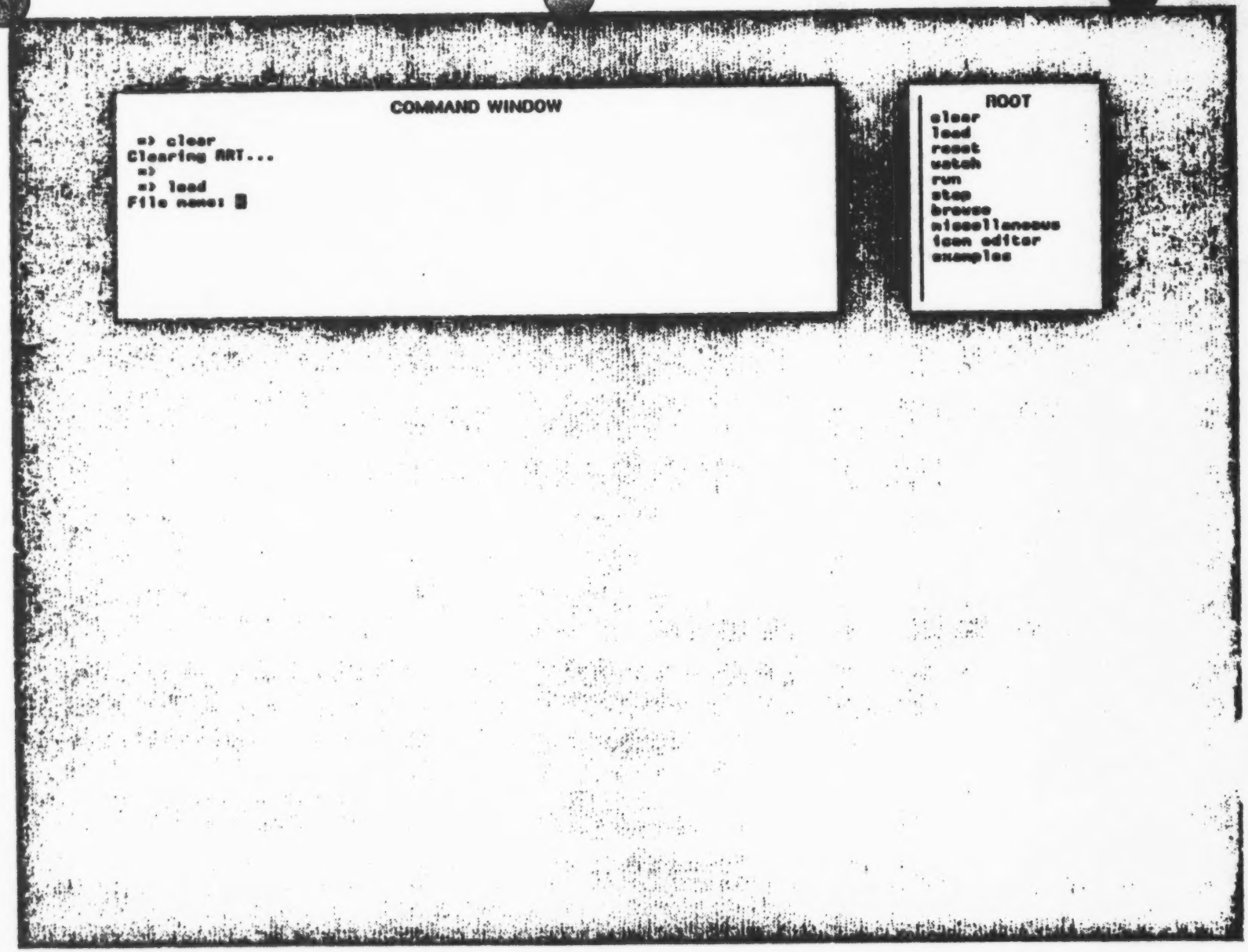
SCREEN 3.1.2.5

ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

SCREEN 3.1.2.7



SCREEN 3.1.2.8.1

```
COMMAND WINDOW
Compiling rule SENSOR-INCONSISTENCY-CHECK-SK... =P=P=J=P=J=P=J=P=J=P=J
=P=J=P+J=P+J=P+J
Loading BIOXIN:>art-test>p-summary.art.1 in package ART-USER and base
10.
Compiling rule PRINTOUT-ALL-FAULTS... =P=J
Compiling rule SUMMARIZE-ALL-FAULTS... =P=P+J
Compiling rule PRINTOUT-REMAINING-FAULTS-FLAG... =P=J
Compiling rule PRINTOUT-REMAINING-FAULTS... =P=P=J=P+J+P+J
 => 
=P+J=P+J=P+J=P+J=P+J
Compiling rule SENSOR-INCONSISTENCY-CHECK-SOURCE... =P=P=J=P=J=P=J=P+J
=P+J=P+J=P+J=P+J=P+J
```

```
ROOT
clear
load
reset
watch
run
step
browse
miscellaneous
icon editor
examples
```

ORIGINAL PAGE IS
OF POOR QUALITY

SCREEN 3.1.2.8.2

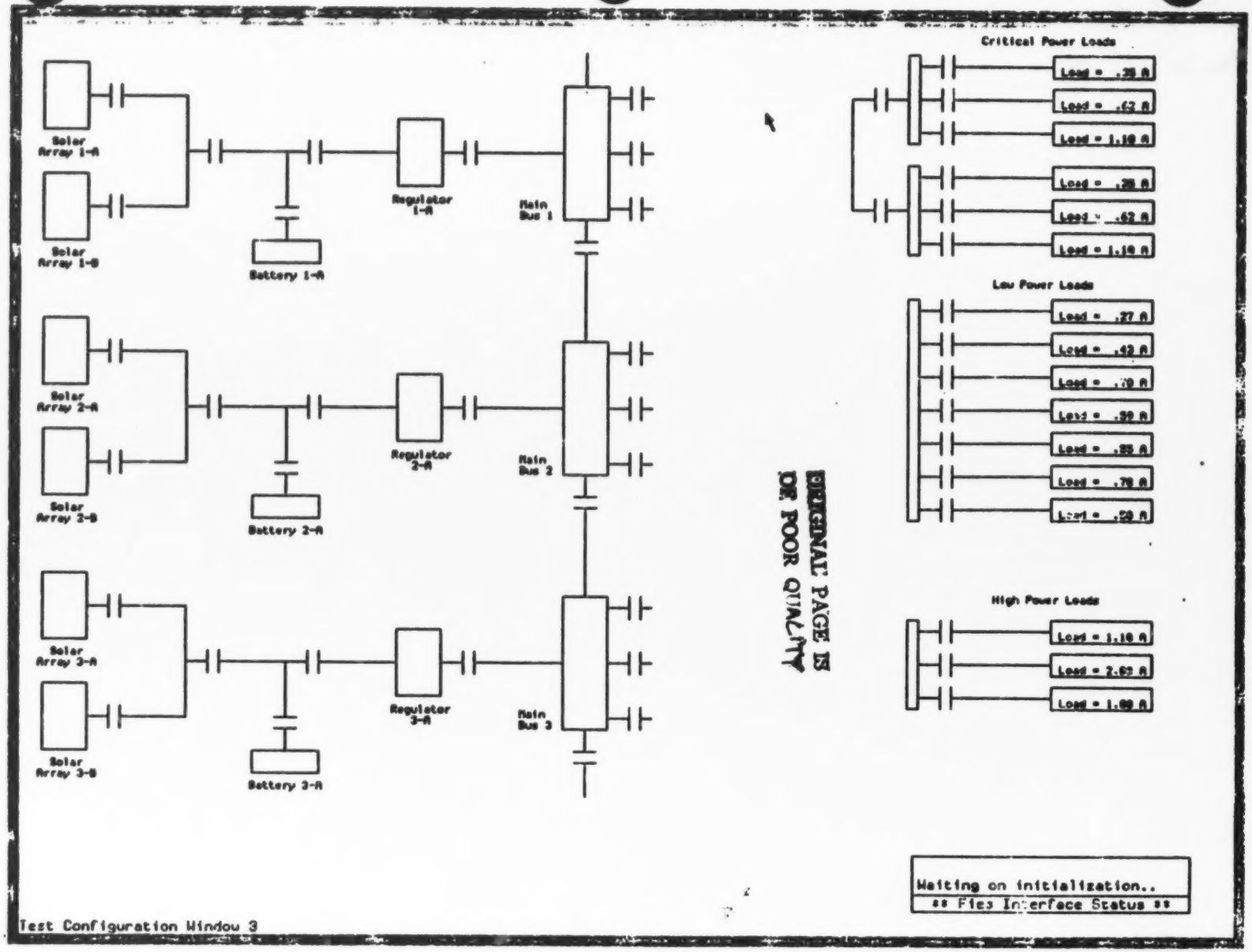
```
COMMAND WINDOW
Compiling rule SENSOR-INCONSISTENCY-CHECK-SK... =P=P=J=P=J=P=J=P=J=P=J
=P=J=P+J=P+J=P+J
Loading DIOXIN:>art-test>p-summary.art.1 in package ART-USER and base
10.
Compiling rule PRINTOUT-ALL-FAULTS... =P=J
Compiling rule SUMMARIZE-ALL-FAULTS... =P=P+J
Compiling rule PRINTOUT-REMAINING-FAULTS-FLAG... =P=J
Compiling rule PRINTOUT-REMAINING-FAULTS... =P=P=J=P+J+P+J
 => reset
Resetting ART...
 => 
=P+J=P+J=P+J=P+J=P+J
```

```
ROOT
clear
load
reset
watch
run
step
browse
miscellaneous
icon editor
examples
```

ORIGINAL PAGE IS
OF POOR QUALITY

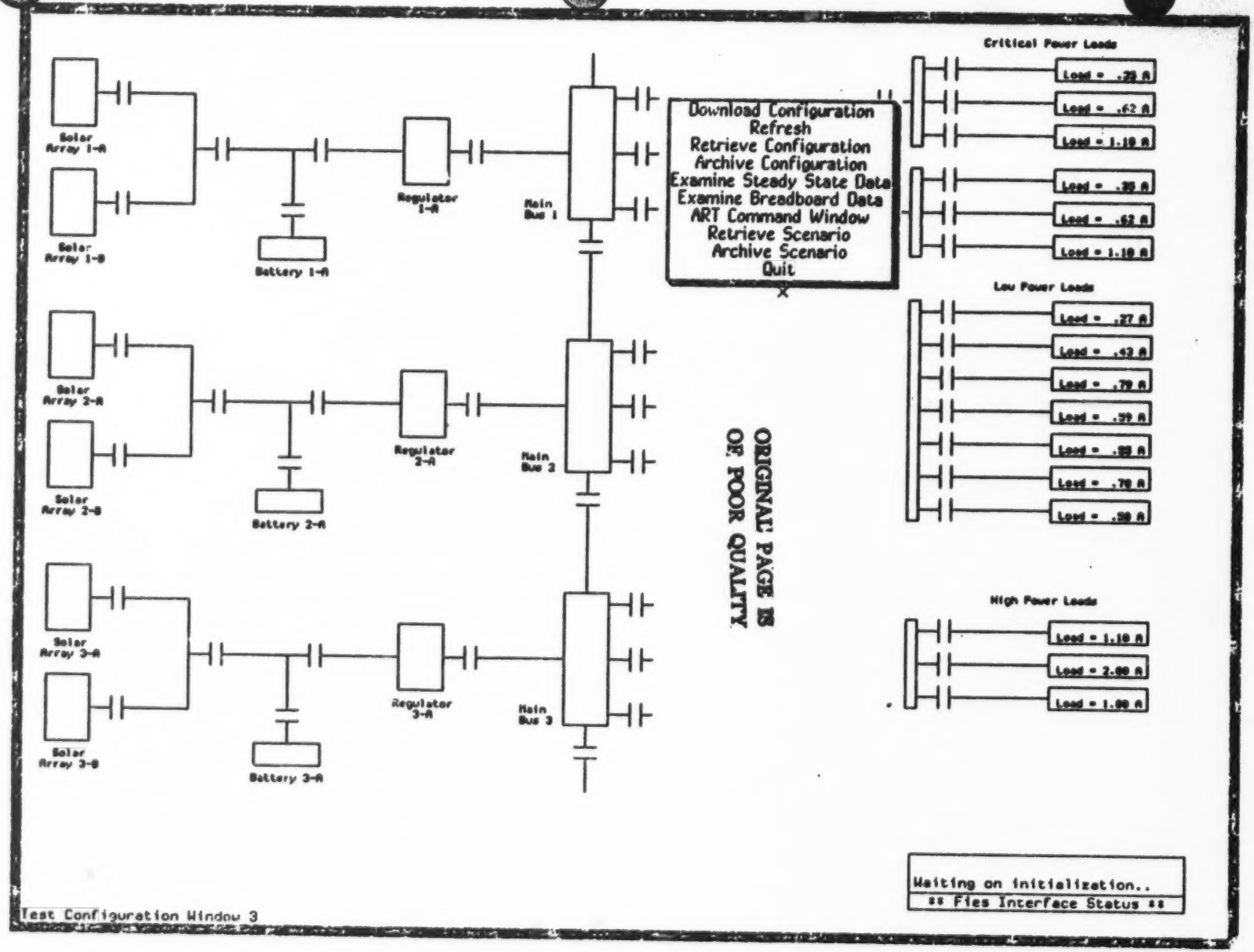
SCREEN 3.1.2.9

PRODUCTION		TEXT	SCREENS
Runtime	Enter Run in ART Window	3.1.3.1	3.1.3.1
	In Symbolics Graphics Display Bring Up Main Menu	3.1.3.2	3.1.3.2
	Select Retrieve Configuration Option	3.1.3.3	3.1.3.3.1, 3.1.3.3.2
	Enter Configuration File Name	3.1.3.4	3.1.3.4
	Observe Configuration in Graphic Display	3.1.3.5	



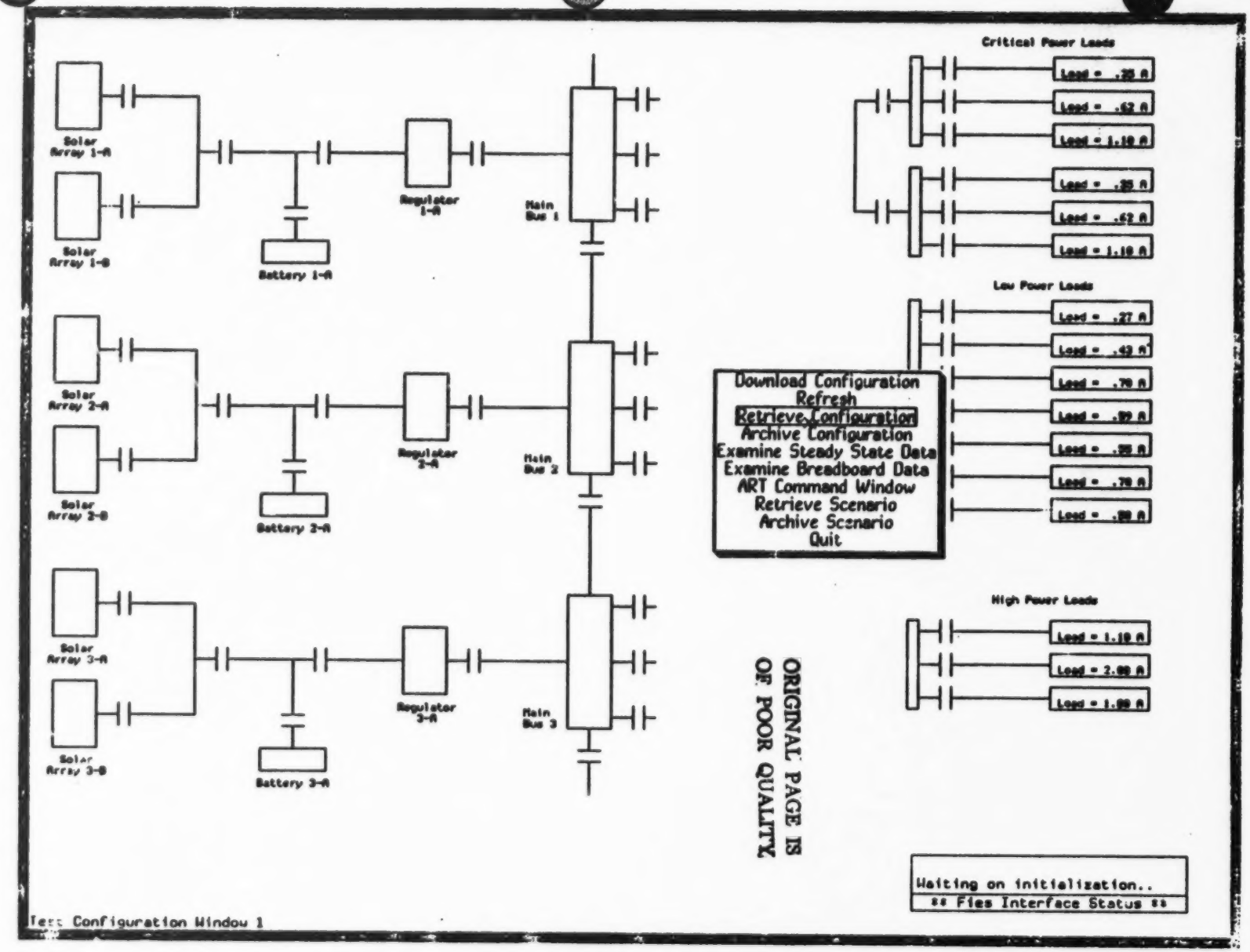
ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 3.1.3.1



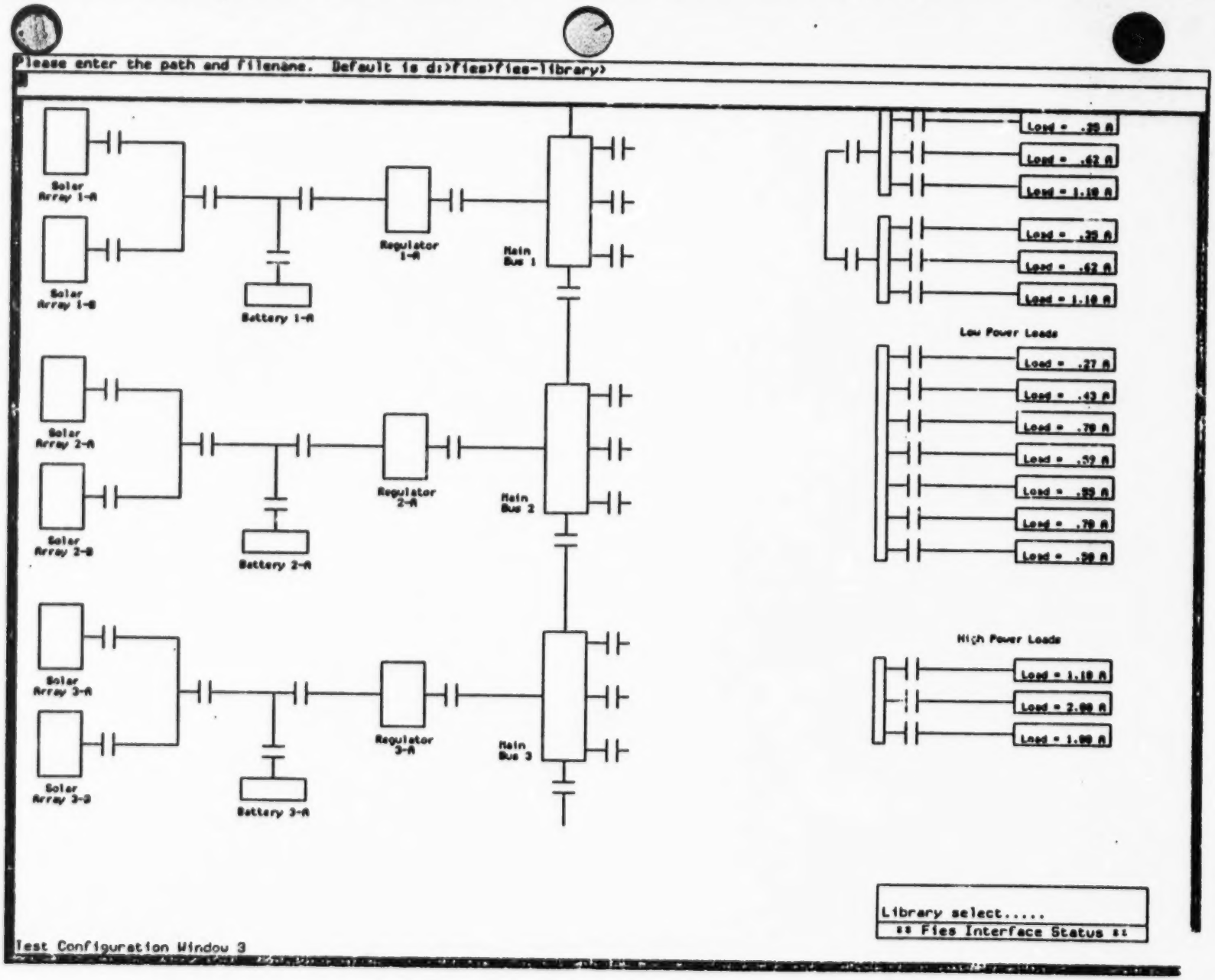
ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 3.1.3.2

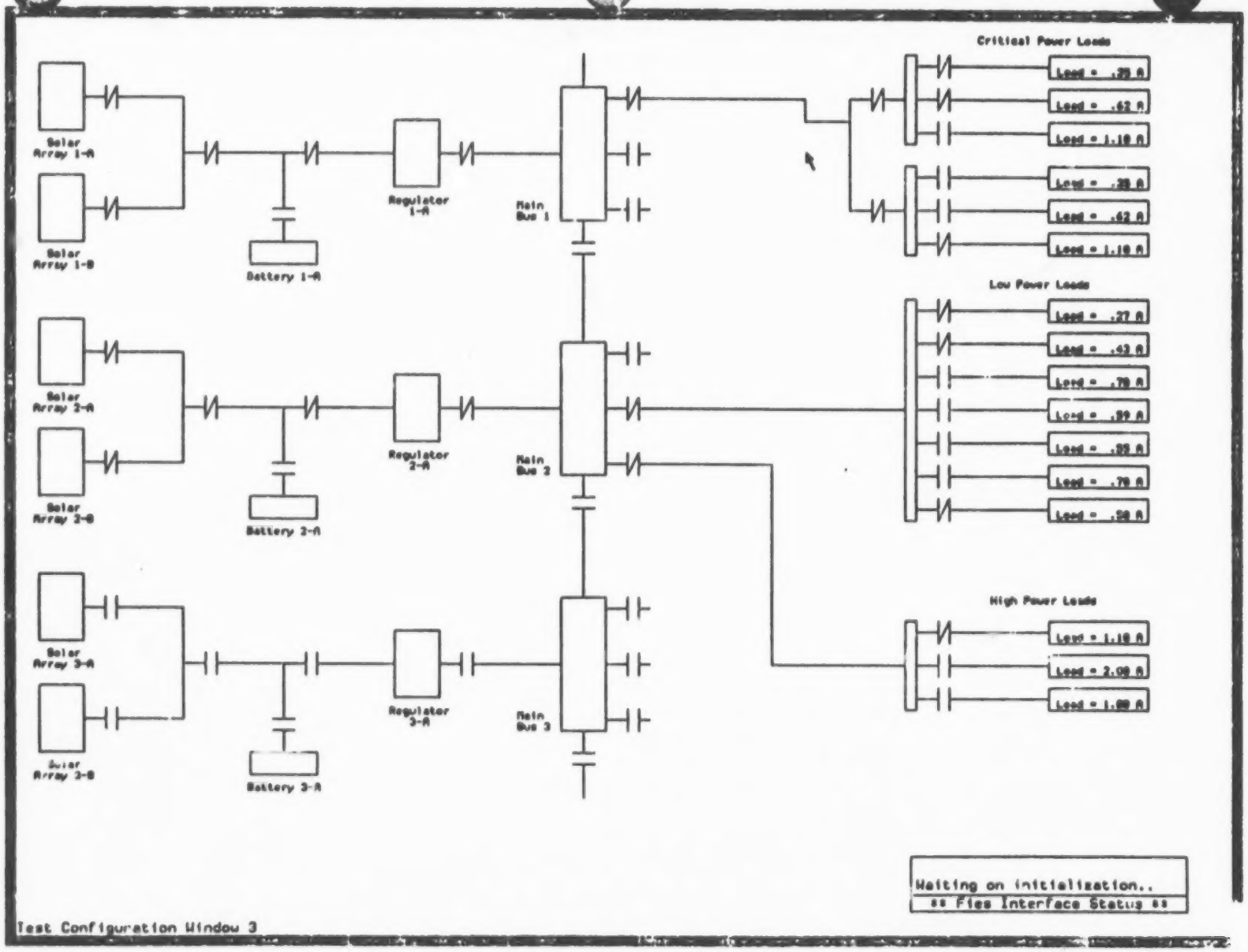


ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 3.1.3.3.1

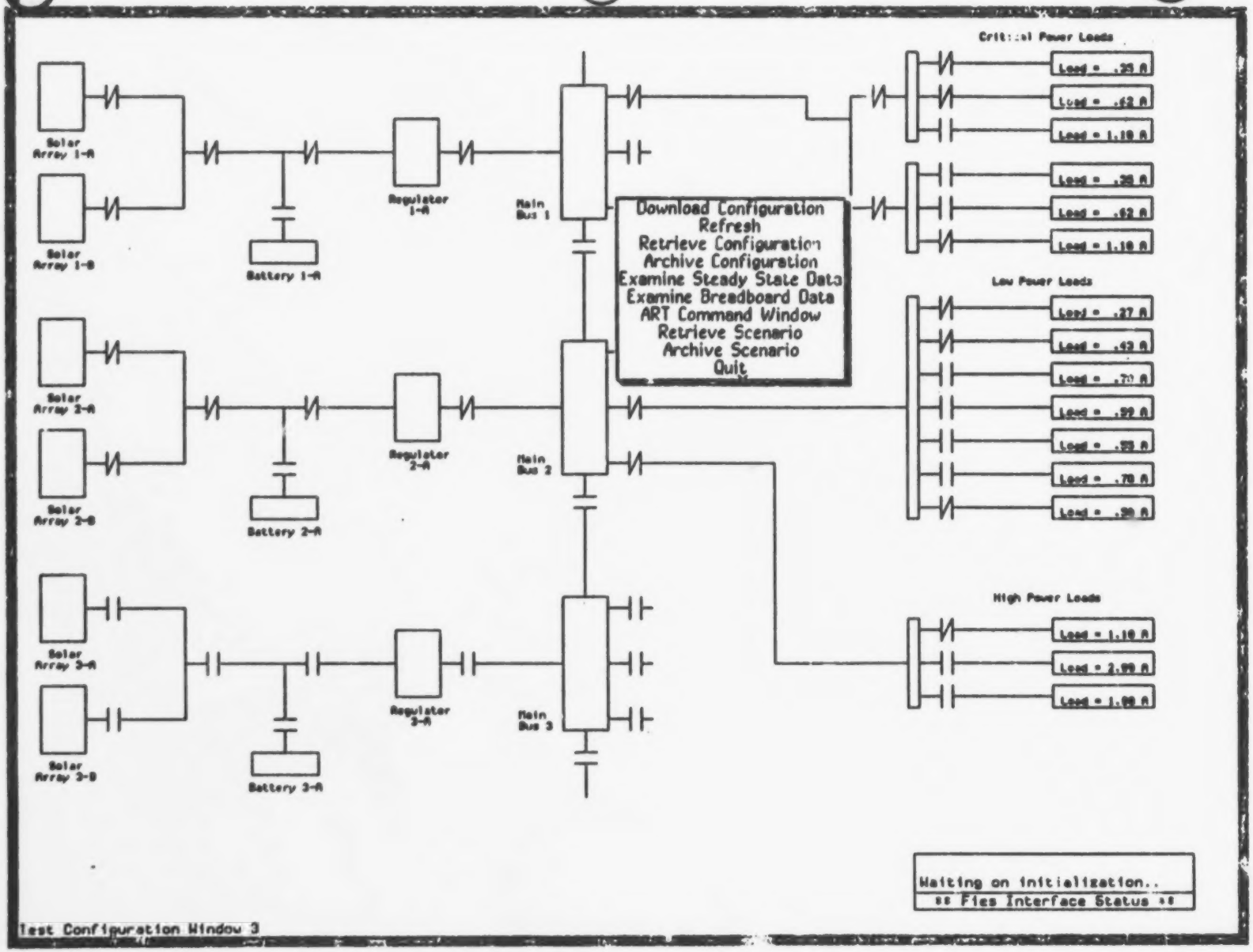


SCREEN 3.1.3.3.2

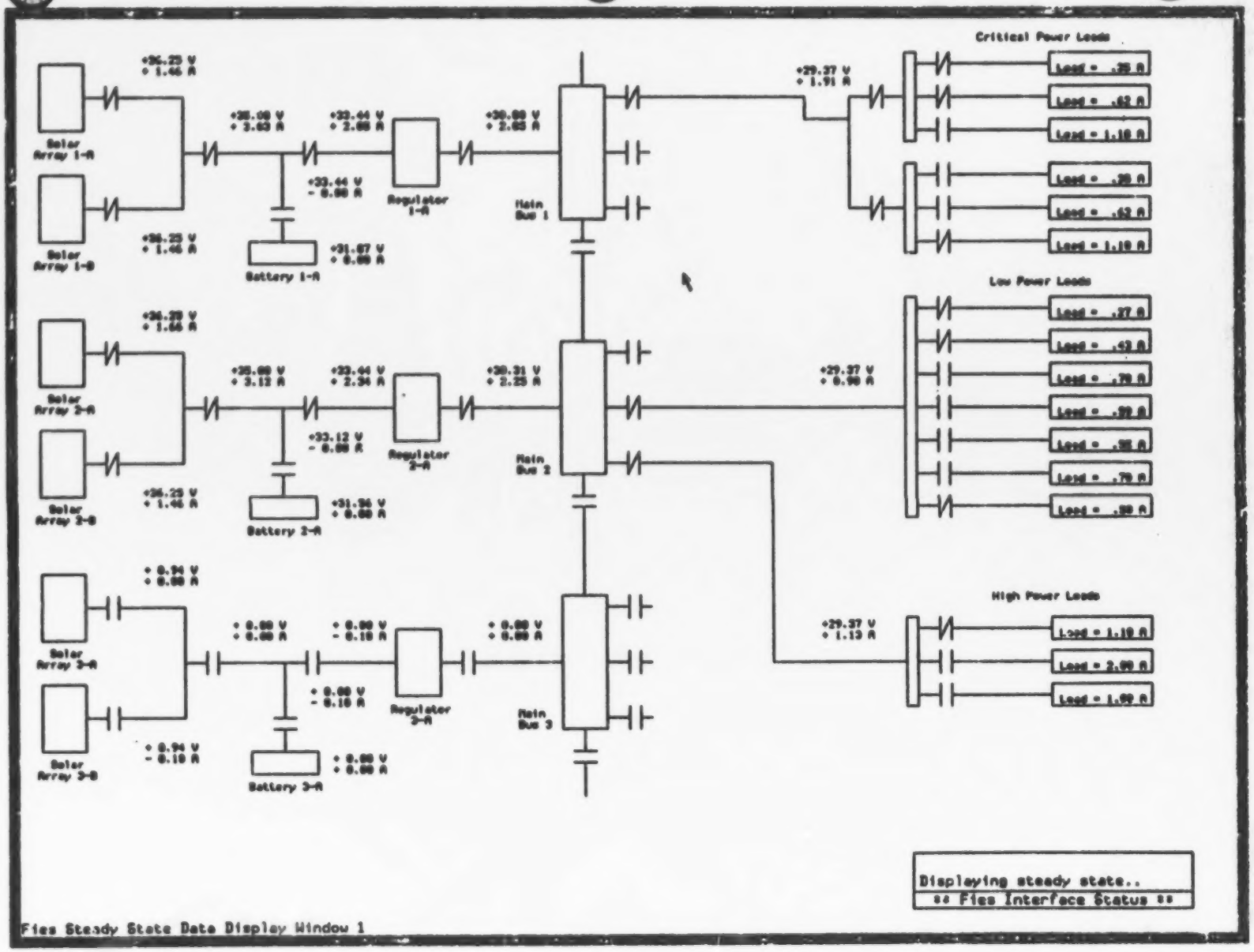

Critical Power Loads
Load = .29 A
Load = .62 A
Load = 1.10 A
Load = .29 A
Load = .62 A
Load = 1.10 A
Low Power Loads
Load = .27 A
Load = .43 A
Load = .79 A
Load = .29 A
Load = .29 A
Load = .79 A
Load = .98 A
High Power Loads
Load = 1.10 A
Load = 2.09 A
Load = 1.00 A
Solar Array 1-A
Solar Array 1-B
Battery 1-A
Regulator 1-A
Main Bus 1
Solar Array 2-A
Solar Array 2-B
Battery 2-A
Regulator 2-A
Main Bus 2
Solar Array 3-A
Solar Array 3-B
Battery 3-A
Regulator 3-A
Main Bus 3
Waiting on initialization..
** Fies Interface Status **
Test Configuration Window 3

SCREEN 3.1.3.4

PRODUCTION		TEXT	SCREENS
Runtime	BRING UP MAIN MENU AGAIN	3.1.3.6	3.1.3.6.1
	SELECT DOWNLOAD CONFIGURATION OPTION	3.1.3.6	3.1.3.6.2
	OBSERVE "I/O WAIT" FAULT DATA MESSAGE ON SYMBOLICS	3.1.3.7	
	INSERT FAULT IN BREADBOARD	3.1.3.8	
	WAIT FOR AND VERIFY RESULTS	3.1.3.9	3.1.3.9



SCREEN 3.1.3.6.1



SCREEN 3.1.3.6.2

COMMAND WINDOW

```
FIRE 418 PRINTOUT-REMAINING-FAULTS f-3188 in (STATE-2)
<== Activation PRINTOUT-REMAINING-FAULTS f-3188 in (STATE-2)
      OPEN-CIRCUIT BUS-2A


FIRE 419 HALT-FIES-END  in (STATE-1)
<== Activation HALT-FIES-END  in (STATE-1)
Program halted.
 =>
      OPEN-CIRCUIT RELAY-2C
```

ROOT

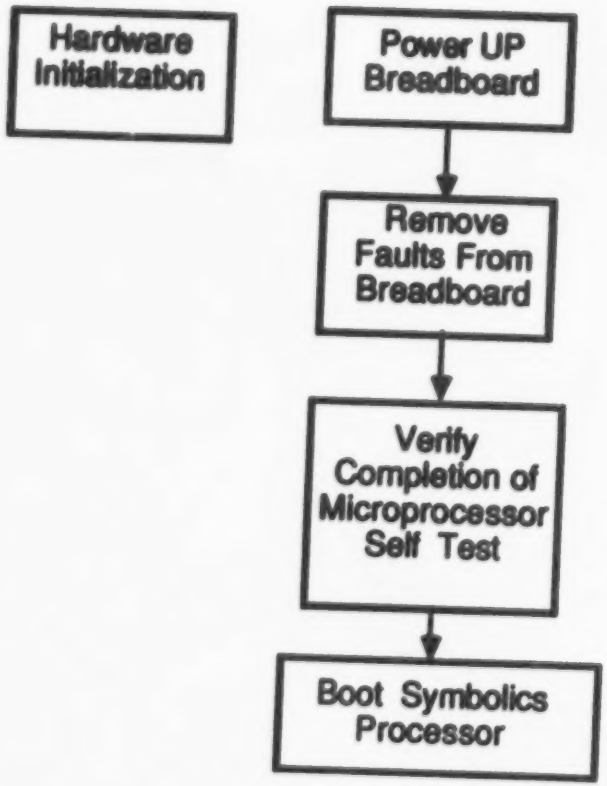
```
clear
load
reset
watch
[redacted]
step
browse
miscellaneous
icon editor
examples
```

ORIGINAL PAGE IS
OF POOR QUALITY

SCREEN 3.1.3.9

INTERACTIVE		TEXT
Overview	Hardware Initialization	3.2.1
	Software Initialization	3.2.2
	Run Time	3.2.3

INTERACTIVE TEXT SCREENS



3.2.1.1

3.2.1.2

3.2.1.3

3.2.1.4 3.2.1.4

Symbolics 3670™ System

This machine is *Martin Marietta Denver Aerospace - Machine Intelligence Lab Einbecker*

Symbolics™ System, Release 6.1
Loaded from FEP0:>mii-r6-1-color.load.1
1536K words Physical memory, 37500K words Swapping space.

Release	6.1
COLOR	136.00
COLOR-[illegible]	68.7
IMAGES	96.21
FEP	137

You are typing to *Lisp Listener 1*. Control characters are
interpreted as commands to edit input. Type Control-[HELP]
for a list of input editor commands.

Use the ":Help" command to display a list of all the Command Processor commands.
Type [SELECT] D to select the Document Examiner to read online documentation.
Type [SELECT] [HELP] for a list of programs.
Type [SYSTEM] [HELP] for a list of asynchronous and window operations.
Click the rightmost mouse button to select the System Menu of programs and window operations.
Type Symbol-[HELP] for a list of special function keys and special character keys.

Please login.
■

(c) Copyright 1985, 1984, 1983, 1982, 1981, 1980, Symbolics, Inc.
All Rights Reserved.

Use the command "Show Legal Notice" to see important legal notices.

Symbolics, Symbolics 3600 and Symbolics 3670 are trademarks of Symbolics, Inc.

Lisp Listener 1

SCREEN 3.2.1.4

INTERACTIVE		TEXT
Software Initialization	Run SDAS.EXE on Microprocessor	3.2.2.1
	Microprocessor Self Test	3.2.2.2
	Microprocessor Reset	3.2.2.3
	Microprocessor Communication Enable	3.2.2.4

INTERACTIVE		TEXT	SCREENS
Software Initialization	Login to Symbolics	3.2.2.5	3.2.2.5
	Select ART	3.2.2.6	
	Clear ART	3.2.2.7	3.2.2.7
	Load FIES Application	3.2.2.8	3.2.2.8.1, 3.2.2.8.2
	Reset ART	3.2.2.9	3.2.2.9

```
(login 'fies-art-interactive)
Loading DIOXIN:>FIES-ART-INTERACTIVE>liepm-init.lisp into package USER
Loading DIOXIN:>fies-art-interactive>make-fies-interface.lisp into package ART-USER
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-interface-defs.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-get-edges.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-clear-status-window-display.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-display-status-message.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-get-configuration-file-name.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-make-configuration-file-name.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-get-scenario-file-name.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-make-scenario-file-name.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-undraw-selected-configuration.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-draw-selected-configuration.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-set-art-relays.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-xmit-relay-configuration.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-set-art-sensors.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-set-art-fault-sensors.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-power-distribution-lines-funcs.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-undraw-power-distribution-lines.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-redraw-power-distribution-lines.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-reset-power-distribution-lines.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-archive-configuration.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-retrieve-configuration.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-archive-scenario.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-retrieve-scenario.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-xmit-request.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-parse-raw-data-for-scenario.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-start-art-scenario.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-send-lisp.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-initialize-mouse.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-initial-configuration.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-serial-interface.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-test-configuration.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-serial-interface-test.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-refresh-data-display.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-serial-interface-level-2.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-steady-state-data-display.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-fault-data-display.bin
Loading DIOXIN:>FIES-ART-INTERACTIVE>fies-display-faults-from-art.bin
T
```

Lisp Listener 1

SCREEN 3.2.2.5

ORIGINAL PAGE IS
POOR QUALITY

COMMAND WINDOW

```
=> clear
Clearing ART...
=>
=> ■
```

ROOT

```
clear
load
reset
watch
run
step
browse
miscellaneous
icon editor
examples
```

ORIGINAL PAGE IS
OF POOR QUALITY

SCREEN 3.2.2.7

COMMAND WINDOW

```
 => clear
Clearing ART...
 =>
 => load
File name: 
```

ROOT

```
clear
load
reset
watch
run
step
browse
miscellaneous
icon editor
examples
```

SCREEN 3.2.2.8.1

COMMAND WINDOW

```
Compiling rule SENSOR-INCONSISTENCY-CHECK-SK... =P=P=J=P=J=P=J=P=J=P=J
=P=J=P+J=P+J=P+J
Loading DIGXIM:>art-test>p-summary.art.1 in package ART-USER and base
10.
Compiling rule PRINTOUT-ALL-FAULTS... =P=J
Compiling rule SUMMARIZE-ALL-FAULTS... =P=P+J
Compiling rule PRINTOUT-REMAINING-FAULTS-FLAG... =P=J
Compiling rule PRINTOUT-REMAINING-FAULTS... =P=P=J=P+J+P+J
 => 
=P+J=P+J=P+J=P+J=P+J
Compiling rule SENSOR-INCONSISTENCY-CHECK-SOURCE... =P=P=J=P=J=P=J=P+J
=P+J=P+J=P+J=P+J=P+J
```

ROOT

- clear
- load
- reset
- watch
- run
- step
- browse
- miscellaneous
- icon editor
- examples

SCREEN 3.2.2.8.2

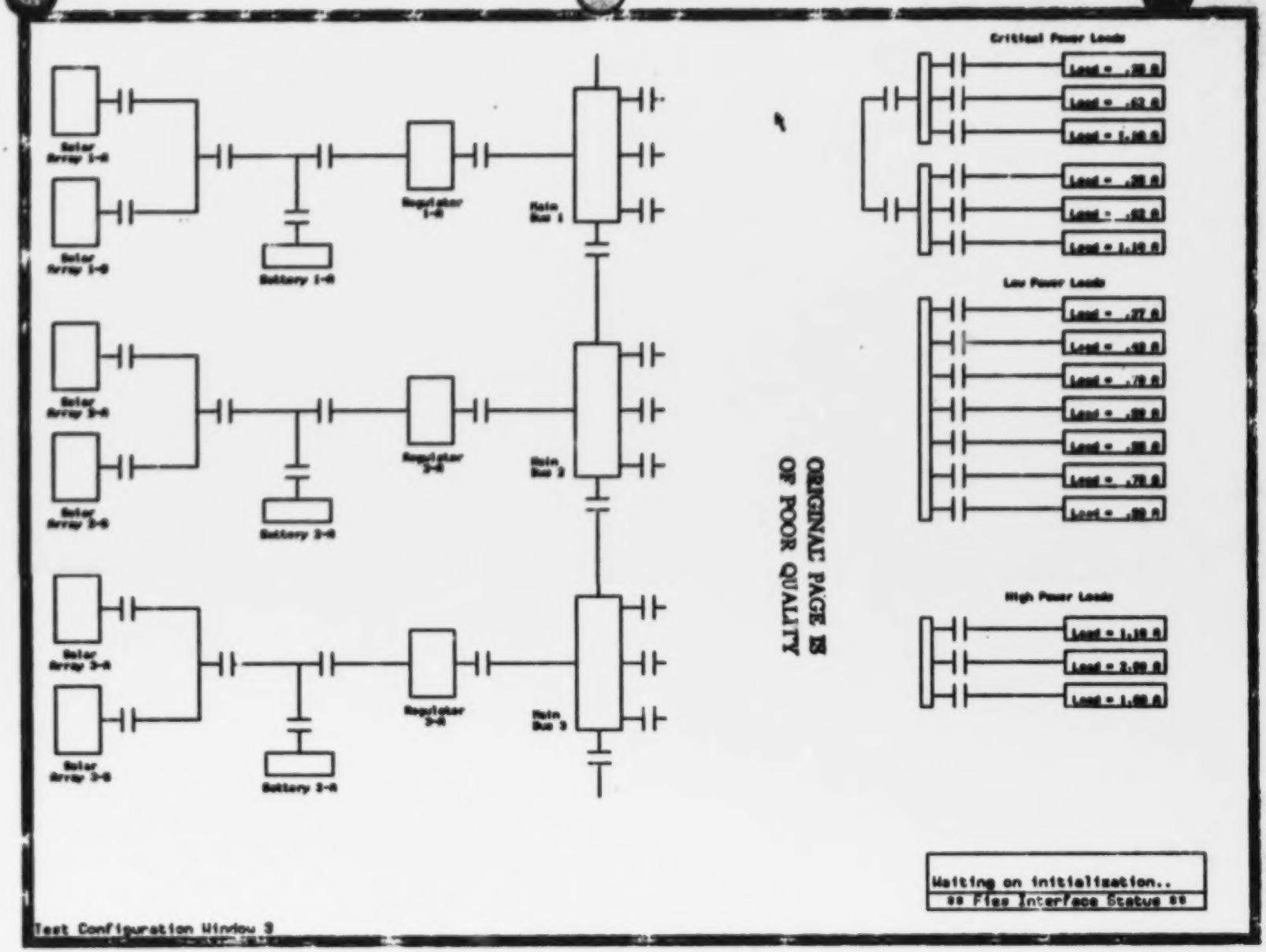
```
COMMAND WINDOW
Compiling rule SENSOR-INCONSISTENCY-CHECK-SK... =P=P=J=P=J=P=J=P=J=P=J
=P=J=P=J=P=J=P=J
Loading D:OXIN:>art-test>p-summary.art.1 in package ART-USER and base
10.
Compiling rule PRINTOUT-ALL-FAULTS... =P=J
Compiling rule SUMMARIZE-ALL-FAULTS... =P=P=J
Compiling rule PRINTOUT-REMAINING-FAULTS-FLAG... =P=J
Compiling rule PRINTOUT-REMAINING-FAULTS... =P=P=J=P=J=P=J
 => reset
Resetting ART...
 => 0
=P=J=P=J=P=J=P=J=P=J
```

```
ROOT
clear
load
reset
watch
run
step
browse
miscellaneous
icon editor
examples
```

SCREEN 3.2.2.9

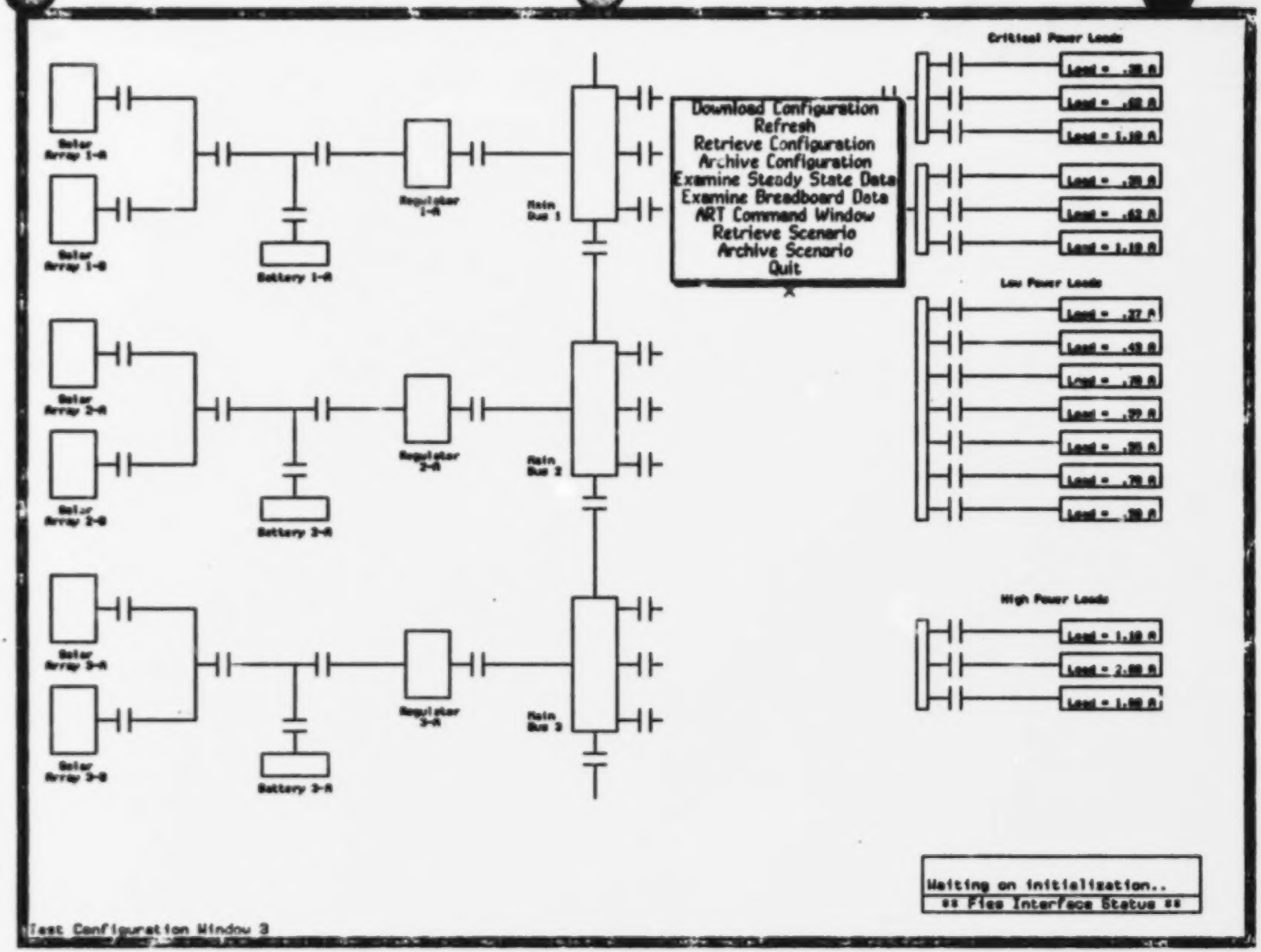
C-2

INTERACTIVE		TEXT	SCREENS
Runtime	Enter Run in ART Window	3.2.3.1	3.2.3.1
	In Symbolics Graphics Display Bring Up Main Menu	3.2.3.2	3.2.3.2
	Select Retrieve Configuration Option	3.2.3.3	3.2.3.3.1, 3.2.3.3.2
	Enter Configuration File Name	3.2.3.4	3.2.3.4
	Observe Configuration in Graphic Display	3.2.3.5	

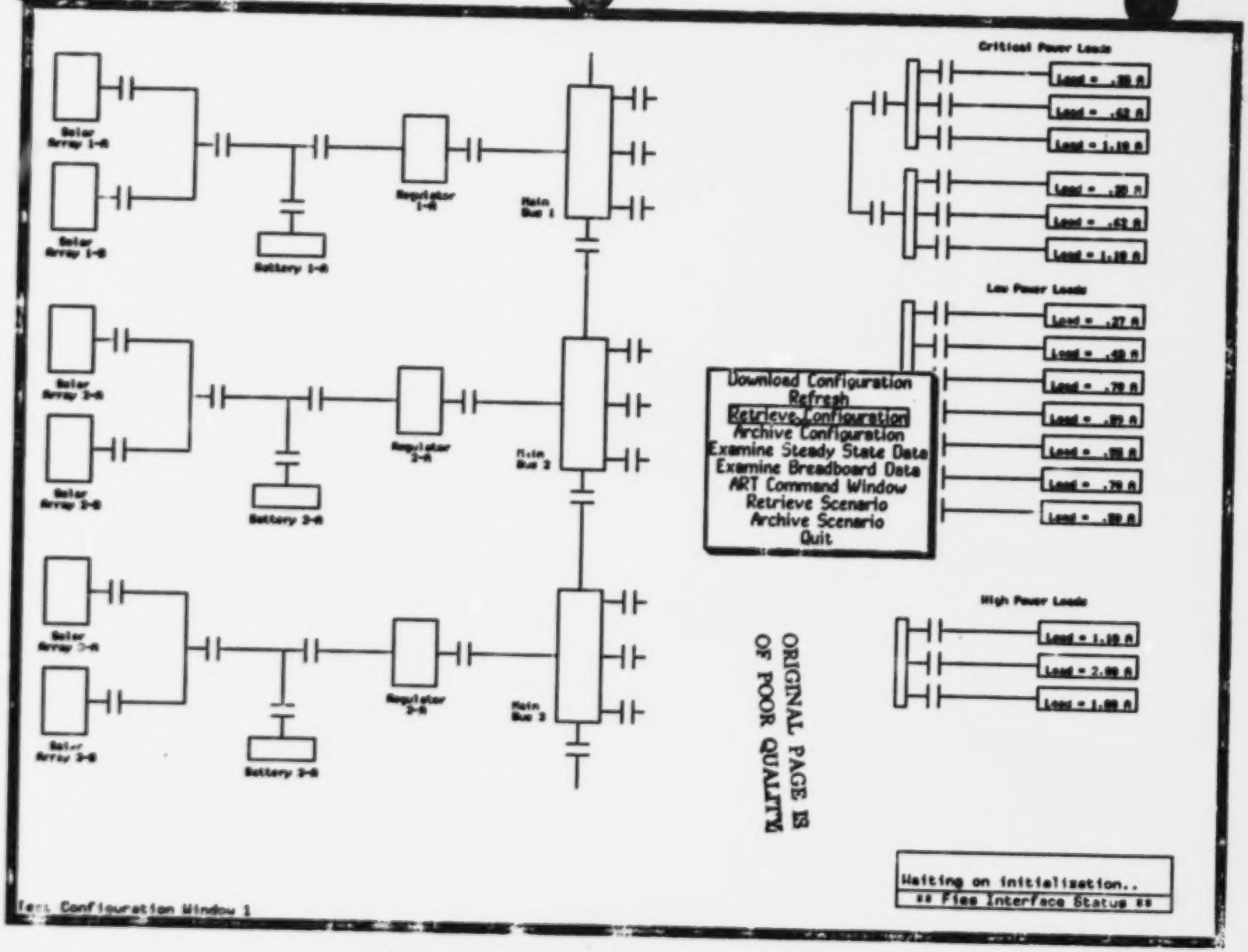


ORIGINAL PAGE IS
OF POOR QUALITY

SCREEN 3.2.3.1

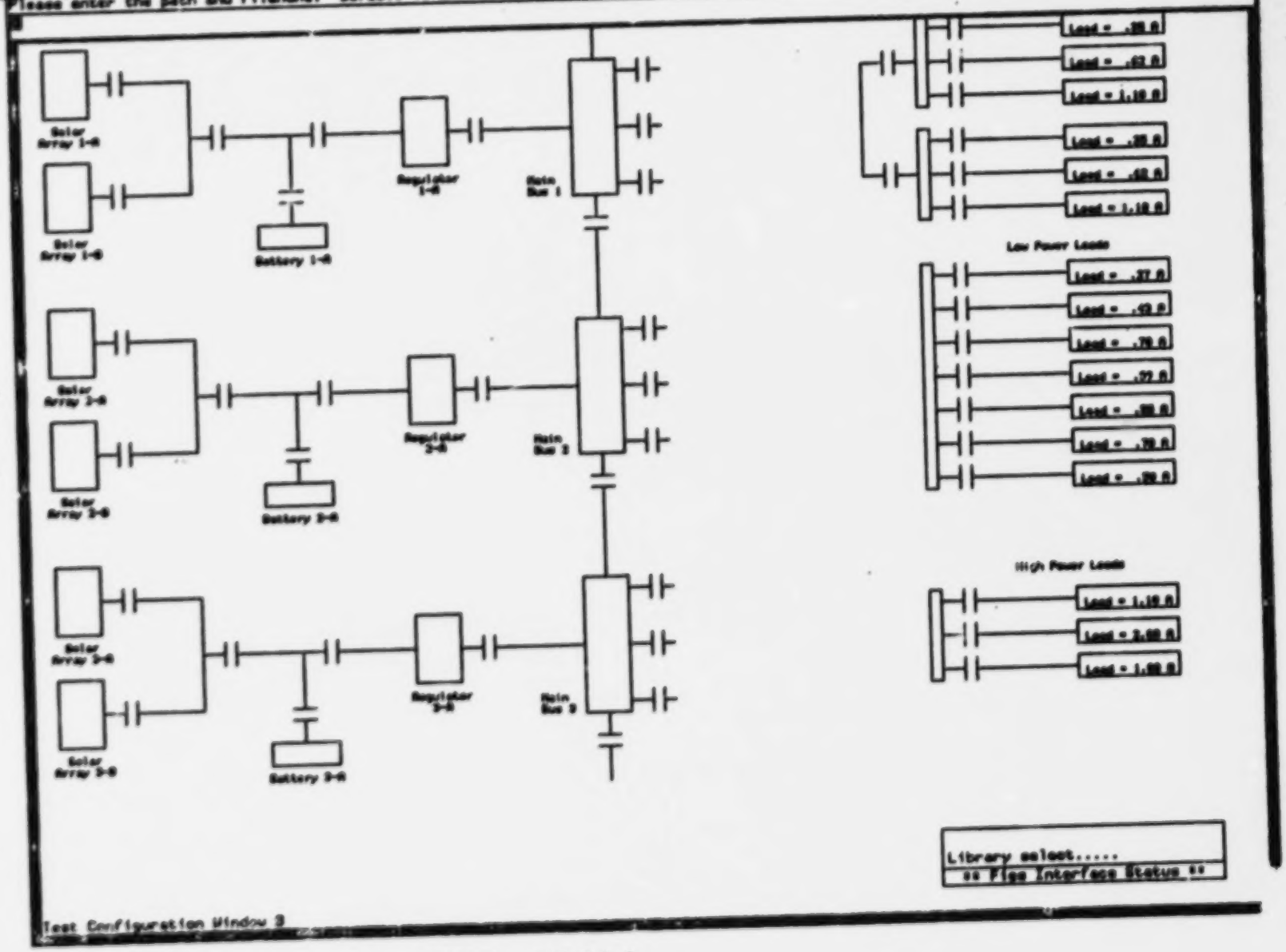

Download Configuration
Refresh
Retrieve Configuration
Archive Configuration
Examine Steady State Data
Examine Breadboard Data
ART Command Window
Retrieve Scenario
Archive Scenario
Quit
Critical Power Loads
Low Power Loads
High Power Loads
Waiting on initialization..
** Flex Interface Status **
Test Configuration Window 3

SCREEN 3.2.3.2

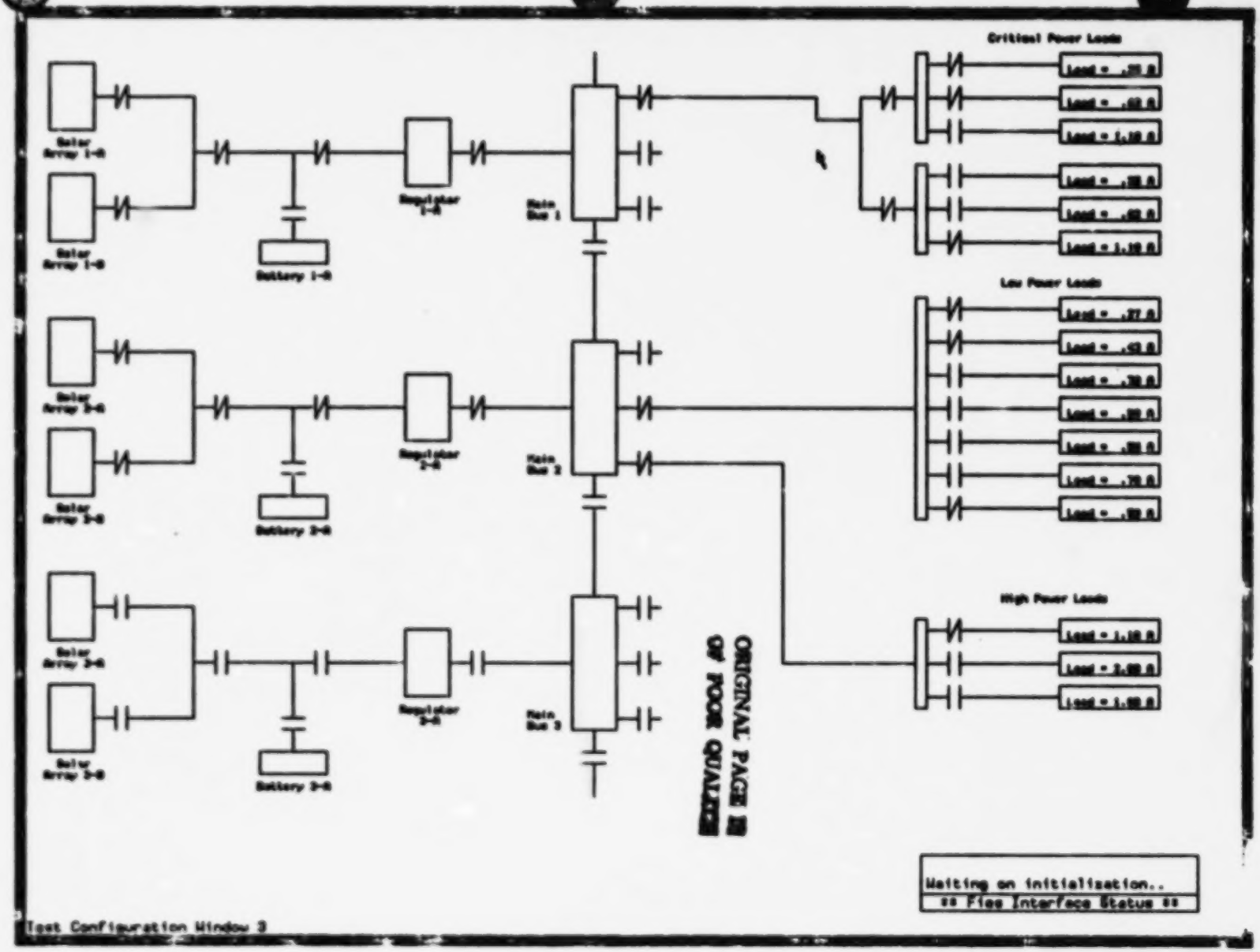


ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 3.2.3.3.1



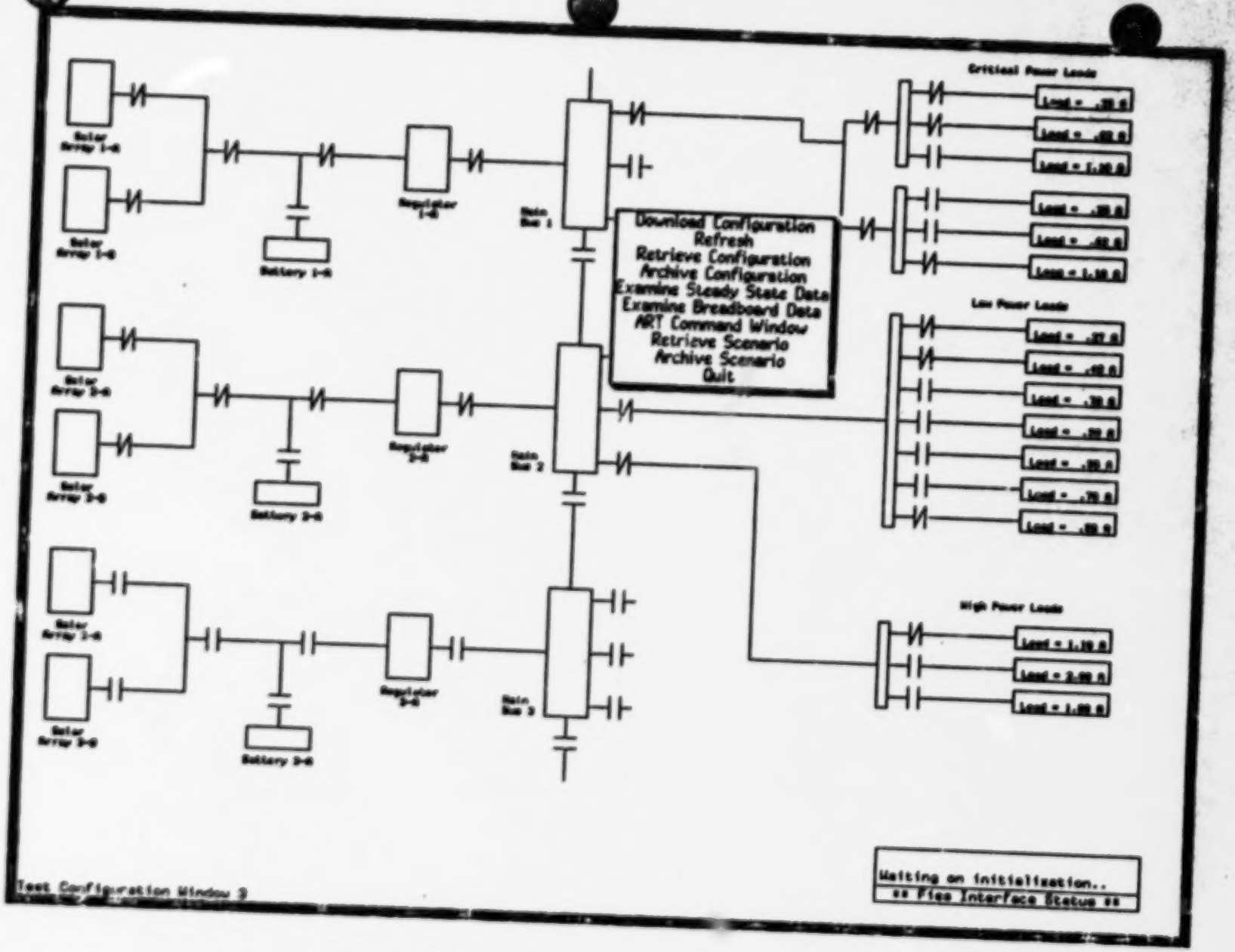
SCREEN 3.2.3.3.2



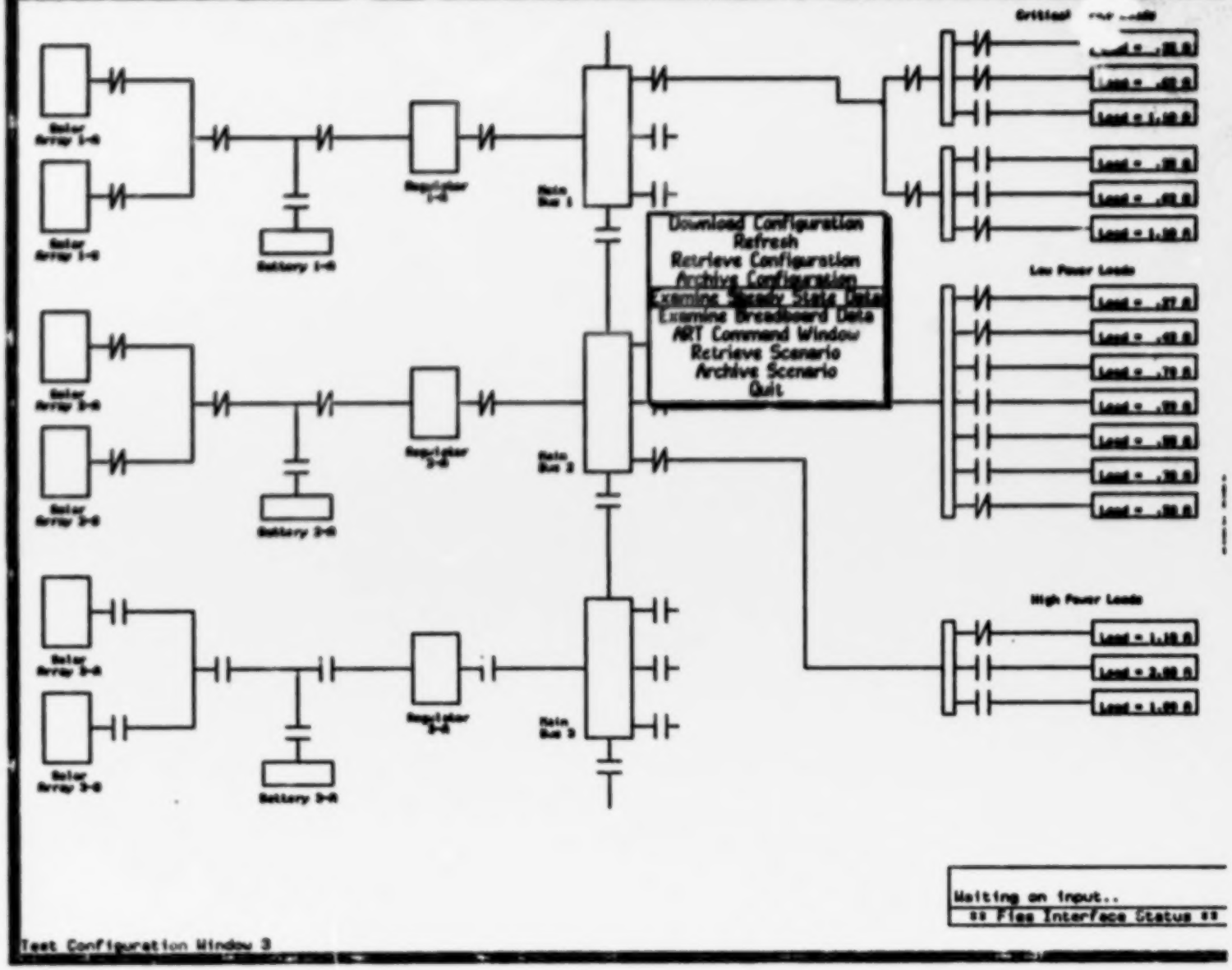
ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 3.2.3.4

INTERACTIVE		TEXT	SCREENS
Runtime	BRING UP MAIN MENU AGAIN	3.2.3.6	3.2.3.6.1
	SELECT DOWNLOAD CONFIGURATION OPTION	3.2.3.6	3.2.3.6.2
	EXAMINE STEADY STATE DATA	3.2.3.6	3.2.3.6.3
	OBSERVE "I/O WAIT" FAULT DATA MESSAGE ON SYMBOLICS	3.2.3.7	
	INSERT FAULT IN BREADBOARD	3.2.3.8	

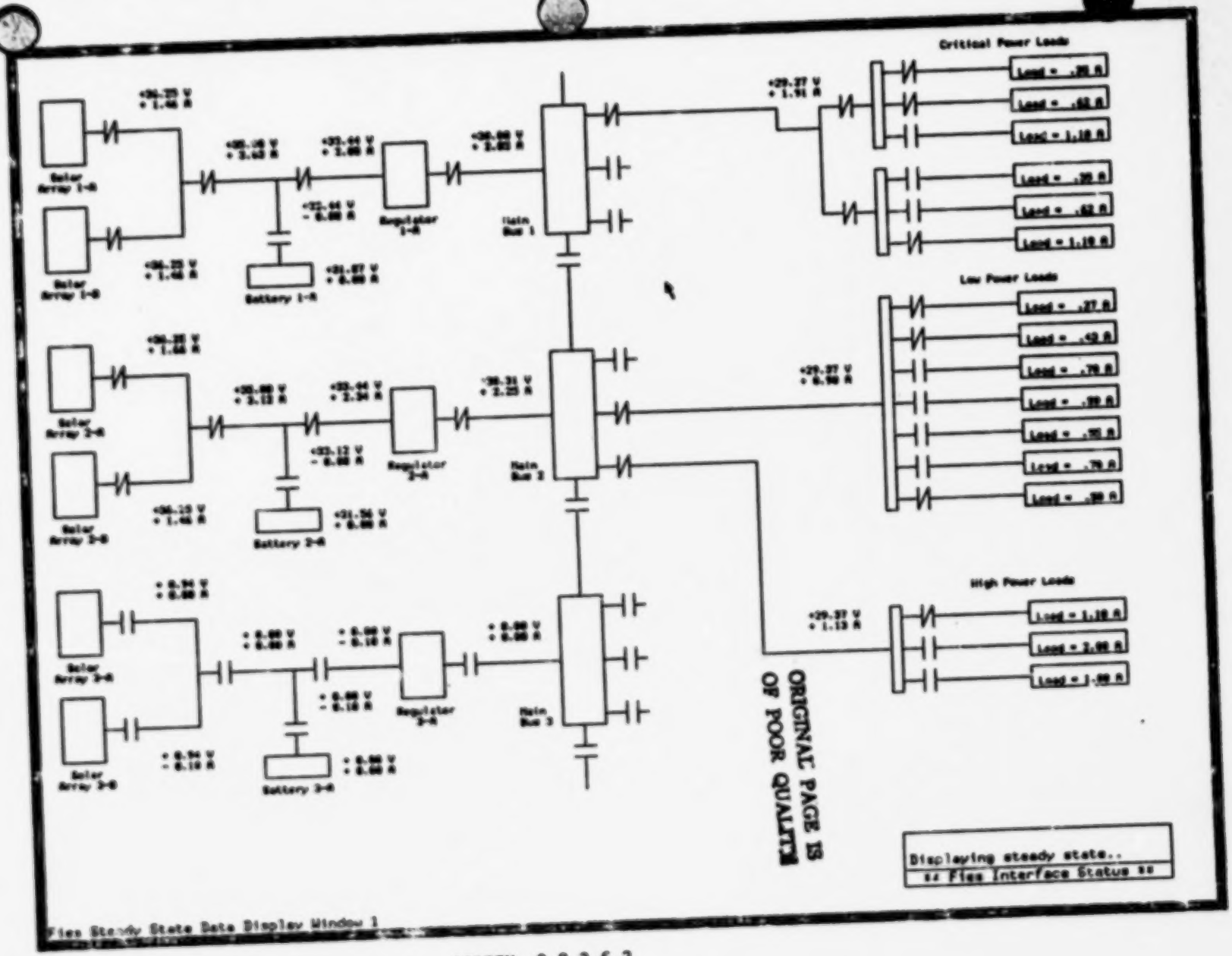


SCREEN 3.2.3.6.1



SCREEN 3.2.3.6.2

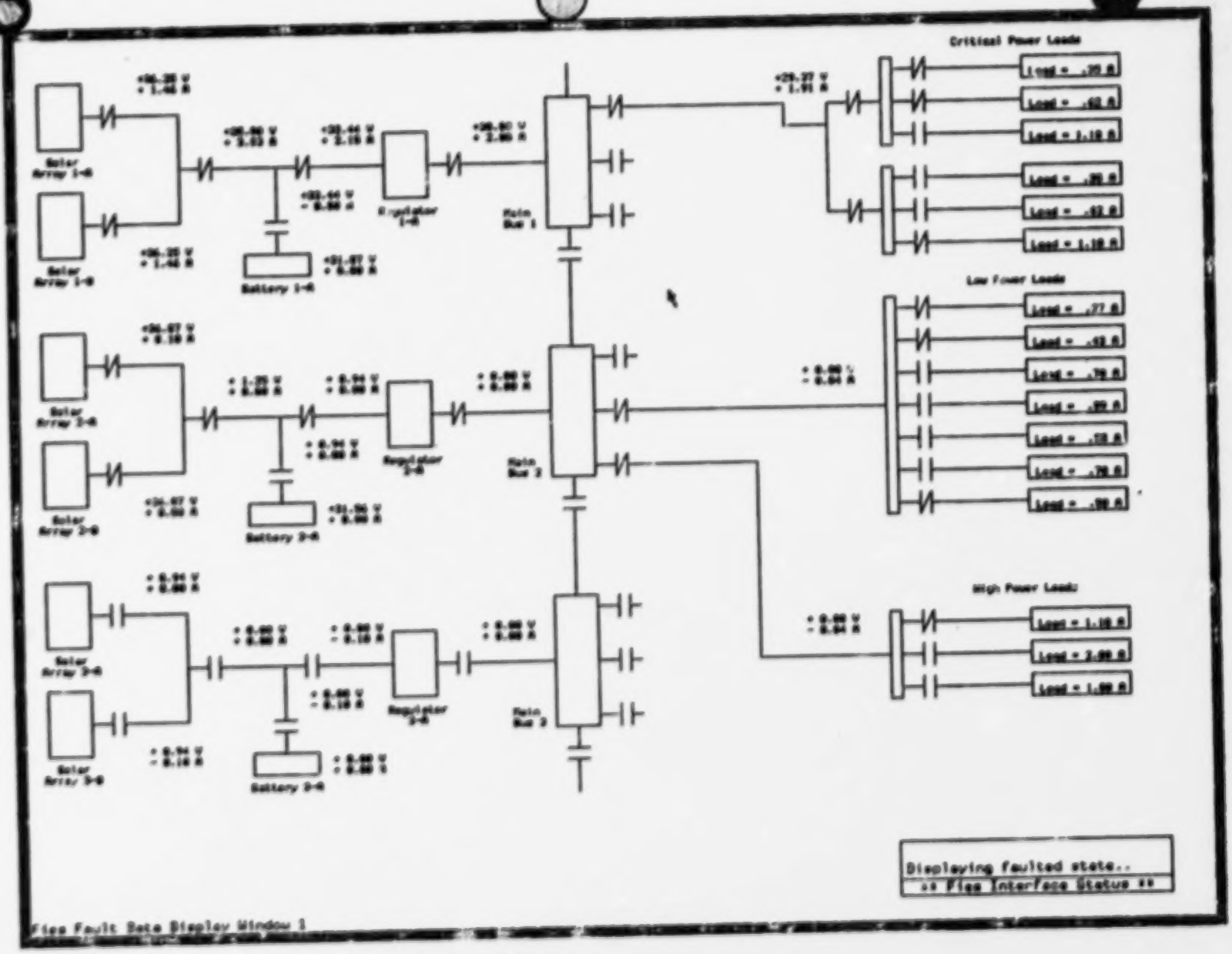
ORIGINAL PAGE IS
OF POOR QUALITY



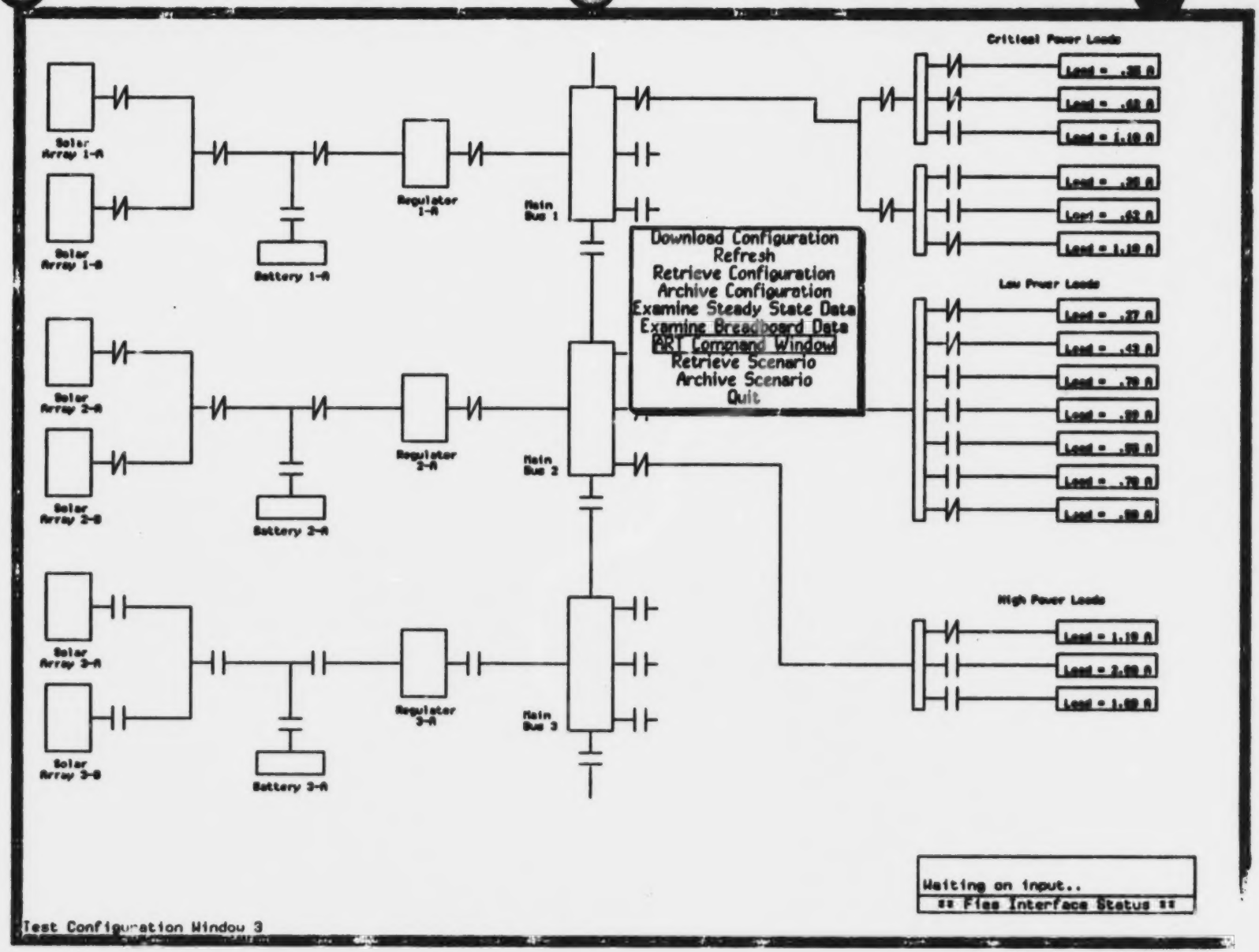
ORIGINAL PAGE IS OF POOR QUALITY

SCREEN 3.2.3.6.3

INTERACTIVE		TEXT	SCREENS
Runtime	EXAMINE BREADBOARD DATA	3.2.3.9	3.2.3.9
	SELECT ART COMMAND WINDOW OPTION	3.2.3.10	3.2.3.10.1
	SET ART WATCH OPTIONS	3.2.3.10	3.2.3.10.2
	RUN ART	3.2.3.10	3.2.3.10.3
	WAIT FOR AND VERIFY RESULTS	3.2.3.10	


Critical Power Loads
Low Power Loads
High Power Loads
Solar Array 1-A
Solar Array 1-B
Solar Array 2-A
Solar Array 2-B
Solar Array 3-A
Solar Array 3-B
Battery 1-A
Battery 2-A
Battery 3-A
Regulator 1-A
Regulator 2-A
Regulator 3-A
Main Bus 1
Main Bus 2
Main Bus 3
Displaying faulted state..
** Flag Interface Status **
Flag Fault Data Display Window 1

SCREEN 3.2.3.9



SCREEN 3.2.3.10.1

ORIGINAL PAGE IS
OF POOR QUALITY

```
COMMAND WINDOW
 =>
=P=J=P+J=P+J=P+J
Loading GIOKIM:>art-test>p-summary.art.1 in package ART-USER and base
10.
Compiling rule PRINTOUT-ALL-FAULTS... =P=J
Compiling rule SUMMARIZE-ALL-FAULTS... =P=P+J
Compiling rule PRINTOUT-REMAINING-FAULTS-FLAG... =P=J
Compiling rule PRINTOUT-REMAINING-FAULTS... =P=P=J=P+J+P+J
 => reset
Resetting ART...
 => run
Program halted.
```

```
ROOT
clear
load
reset
run
step
browse
miscellaneous
icon editor
examples
```

ORIGINAL PAGE IS
OF POOR QUALITY

SCREEN 3.2.3.10.2

COMMAND WINDOW

```
FIRE 418 PRINTOUT-REMAINING-FAULTS f-3188 in (STATE-2)
<== Activation PRINTOUT-REMAINING-FAULTS f-3188 in (STATE-2)
      OPEN-CIRCUIT BUS-2A

FIRE 419 HALT-FIES-END  in (STATE-1)
<== Activation HALT-FIES-END  in (STATE-1)
Program halted.
 =>
      OPEN-CIRCUIT RELAY-2C
```

ROOT

```
clear
load
reset
watch
[illegible]
step
browse
miscellaneous
text editor
examples
```

ORIGINAL PAGE IS
OF POOR QUALITY

SCREEN 3.2.3.10.3

APPENDIX B

Fault Isolation Expert System Test and Demonstration Plan

Fault Isolation Expert System (FIES)
Test and Demonstration Plan

0.0 PURPOSE

The purpose of the FIES Test and Demonstration Plan, as the name suggests, is twofold. First it discusses the considerations, scope, and procedures for verifying the operational status and system capabilities of FIES. And secondly, it presents a scenario for the final acceptance demonstration at the Marshall Space Flight Center.

1.0 FIES DESCRIPTION

FIES consists of a number of integrated hardware and software subsystems. The overall "health" of FIES is determined by the combined operational status of these subsystems. The strategy behind testing FIES operational status is based on testing "first things first". A brief description of the subsystems is presented below.

1.1 FIES Hardware

FIES hardware is divided into three major subsystems. The first is the Breadboard which contains solar array simulators (power supply modules), batteries, network configuration relays, load simulation modules, measurement networks, and control logic. These components may be configured into a power subsystem (also referred to as a power network) manually at the Bread Board Fault Insertion Panel, via keyboard entry at the Micro Processor Terminal (Monitor), or through the Graphics/Mouse Interface on the SYMBOLICS Terminal. Four fault types are supported and may be inserted via the Bread Board Front Panel Fault Insertion switches. The fault types include:

'opened relay'	-	which interrupts current through some point in the power subsystem,
'closed relay'	-	which allows fault current to flow to some point in the power subsystem,
'resistive shunt'	-	which causes excessive current drain (1-2 amperes) at some point in the power subsystem, and
'direct shunt'	-	which causes an infinite current sink at some point in the power subsystem that is limited only by the current sourcing capability of the power modules.

The second major hardware subsystem is the Micro Processor. The Microprocessor provides an operating system for managing and monitoring the Bread Board Subsystem. A bit oriented memory map allows individual control over the network configuration relays. Writing a '1' to the corresponding memory bit forces a relay contact closure and writing a '0' forces the relay contacts open. In this way, various power networks can be dynamically configured for evaluation and demonstration of the fault isolating capabilities of FIES.

The Microprocessor provides hardware for monitoring the voltage and current levels at 24 sensor node locations in the power subsystem. It is also equipped with added I/O capabilities for interfacing to the Bread Board Front Panel, the Monitor Terminal, and the SYMBOLICS machine.

The third hardware subsystem is the SYMBOLICS machine. The SYMBOLICS provides the processing environment for the Expert System software, which is an extension of the Automated Reasoning Tool (ART). Graphic capabilities supported by the SYMBOLICS terminal provides a user environment that is easy to master.

1.2 FIES Software

The software is divided into three main subsystems (or Layers). The layer closest to the Bread Board is the Sensor Data Acquisition Scheduler (SDAS) whose task is to coordinate the activities of the Micro Processor. These activities include controlling the Bread Board hardware, monitoring the sensor measurement nodes for steady or fault states, and handling communications with the SYMBOLICS.

The middle layer, the Expert System Communications Interface, resides on the SYMBOLICS. Its activities include communicating with SDAS (on the Micro Processor), supervising the sensor load network configuration, managing the sensor load measurement data, interacting with Human Operators, and interfacing to the Expert System.

The outermost layer is the Expert System itself. Its activities include interacting with the Expert System Interface and hypothesizing the source of faults manually introduced via the Fault Insertion Panel on the Bread Board.

2.0 FIES Testing Strategy

FIES testing is clearly divided into two areas of effort. The first area involves verify that the hardware and software system are working normally. In this context, 'normally' means that the hardware is operating within its normal limits and that the software is basically 'sane'. This portion of the testing effort will hereafter be referred as "FIES Operational Status Testing".

The second area of testing involves exercising the fault isolating capabilities of the Expert System software. This testing effort assumes that the operational status of FIES has been verified and that all faults detected correspond to faults manually inserted at the Bread Board Fault Insertion Panel.

2.1 FIES Operational Status Testing

Testing of FIES operational status is approached from a "layered" point of view and will be done in the following four phases:

1) The Bread Board Hardware Integrity Test (Procedure A) verifies that the fault insertion and sensor load hardware are operational.

2) The Microprocessor Software Integrity Test (Procedure B) verifies that the Sensor Data Acquisition Scheduler (SDAS) is running normally, that sensor node voltage/current levels can be measured accurately, and that steady and fault states can be achieved.

3) The Microprocessor - SYMBOLICS Communication Subsystem Test (Procedure C) verifies that control handshaking and sensor node data transferral between the SYMBOLICS and Micro Processor are operational.

4) The SYMBOLICS - FIES Operational Test (Procedure D) verifies that the FIES Software is operational and capable of hypothesizing a single fault correctly. (This test is a simple sanity test; its intention is not to exhaustively prove the fault isolating capabilities of FIES.)

2.2 FIES Capability Testing

While the previous tests verify the basic operational status and integrity of FIES, they do not attempt to fully exercise the reasoning and fault isolation capabilities of the underlying Automated Reasoning Tool (ART) software upon which FIES is built. For that matter, no test will be able to verify all possible permutations of the FIES sensor load network. The best that can be done is to cover as many configuration/fault combinations as is reasonable and yet still practical.

Some important considerations come to mind when choosing a benchmark set of configuration/fault combinations. FIES should be able to isolate faults in small, minimally loaded configurations as well as more complicated, heavily loaded configurations. It should be able to deal with single bus configurations as well as multi bus arrangements. All four supported fault types, `open relay', `closed relay', `resistive shunt', and `direct shunt', should be applied in various test cases but for at least one complete multiply-loaded bus configuration, all fault types should be applied at each and every point in the sensor load network.

Test Procedure E presents a test procedure for generically testing the fault isolating capabilities for any configuration/fault combination.

3.0 Demonstration Strategy

This section outlines the procedure and context for the demonstration performed at MSFC.

3.1 Content

The demonstration consists of a selected number of test uses. This set is a subset of the cases used for the formal acceptance test. These cases represent a range of network configurations, ranging from single sink, single source configurations to multiple sink multiple source configurations. These cases were chosen using 3 criteria:

1) A case must be representation of a possible power system configuration.

2) A case must not cause a power overload in the system.

3) A case must not violate any of the conditions described in chapter 6.

3.2 Procedure

3.2.1 Initialization

The computer system was initialized by MMC personnel. This included all phases of integrity testing described in procedures A-E.

3.2.2 Demonstration

The rational for each case (based on the criteria of section 3.1) was presented. Following this presentation on each case MSFC personnel executed the FIES system for the case. Discussion as required followed.

Breadboard Hardware Integrity Test
(Procedure A)

1.0 o With power cord unplugged, use an ohmmeter to verify that no short exists between the hot and neutral AC lines. If non exists, plug the AC power cord into a 115VAC, 10AMP circuit.

2.0 o With the circuit breaker in the 'off' position, verify that +115AC is present between the "neutral" and the "line" terminals of the CORNELL-DUBILIER line filter.

3.0 o Apply power to the breadboard by throwing the circuit breaker to the 'ON' position. Verify that the "digital subsystem" and the "power subsystem" lamps are glowing.

4.0 o Measure the voltage between the + and - sense terminals on the H.P. 6264B power supply. This voltage should be +5VDC, ±5%. Verify that the current display is reading approximately 10 amperes of current load.

5.0 o Measure the voltage between the + and - sense terminals of the 6 power modules contained in the 3H.P. 6255A power supplies. This voltage should be + 38.5 VDC, ± 2%. With the meter switch set to the 1.8A scale, verify that the current load is presently reading 0 amperes.

6.0 o Verify that the "boot" procedure has completed by observing the "-" prompt on the system console. Enter "SDAS.EXE" and a carriage return on the system console.

7.0 o Place the rotary switch in the "status" position. Place the node I.D. thumbwheel switch to '22'. Depress the "Repeat Function" momentary switch on the main panel. Observe that the node 22 voltage reads +33.75VDC, ±5% in the alphanumeric display. This indicates that the power module 1 battery is in place, and appears to have a sufficient charge to drive the breadboard.

8.0 o Set the Node I.D. to 23. Again depress the "Repeat Function" switch and observe that the module 2 battery voltage is +33.75VDC, ±%5.

9.0 o Set the Node I.D. to 24. Depress the "Repeat Function" switch and observe that the module 3 battery voltage is +33.75VDC, ±5%.

10.0 o Set the two paddle switches on the Fault Insertion panel to "closed" and "resistive", respectively.

11.0 o Raise toggle switches 1, 4, 5, 6, 22, 27, and 30. Verify that all other toggle switches are in the center position. Verify that the "Low Power", "High Power", and "Critical Power" lamps are glowing. This verifies that power module #1-A is supplying power to the system.

12.0 o Set toggle switch #1 to the center position and verify that the "Low Power", "High Power", and "Critical Power" lamps are extinguished. Raise toggle switch #2 and verify that the three lamps are again illuminated. This verifies that power module #1-B is supplying power to the system.

13.0 o Lower toggle switch #2 to the center position and verify all three of the power load branch lamps go out. Raise toggle switch #3. Verify that the three lamps are again illuminated. This assures that the battery in module 1 is supplying power to the system. Return all toggle switches to the center position.

14.0 o Raise toggle switches 7, 10, 11, 12, 23, 25, and 28. Again all three load branch lamps should be illuminated. Returning toggle switch #7 to the center position should cause all three lights to go out.

15.0 o Raise toggle switch #8, verifying that the three load branch lamps are re-powered. Returning #8 to center position extinguishes the lamps. If all three lamps are again lit when switch #9 is raised, then the battery in module #2 has sufficient power to drive the network. Return all relays to center position.

16.0 o Raise toggle switches 13, 16, 17, 18, 24, 26, 29. As before the three load branch lamps should be lit. Returning #13 to center position and then raising #14 should cause the lamps to momentarily go out. Lowering #14 and then raising #15 should have the same effect. Return all toggle switches to center position.

17.0 o The breadboard is now ready for microprocessor software testing.

Micro Processor Software Integrity Test
(Procedure B)

1.0 o Perform and verify Micro Pro hardware test

2.0 o Boot micropro and verify startup diagnostics: (Refer to INTEL's Software Diagnostic Test Manual if error)

3.0 o Load and start SDAS.EXE

4.0 o Put the bread board rotary switch in `TEST MODE'. Bread board will respond with "MONITOR SRC MODE" on ASCII Display Device. This action will cause the MONITOR keyboard to be substituted for the SYMBOLICS (at the communication interface level only).

o Care must be taken when entering command strings in this mode because the format is fixed and the Micro Pro software is not very forgiving (a misformed command string can hang the Micro Pro thus requiring a reboot). (A <BACKSPACE> character can be entered to discard previously mistyped characters.)

o Note that command strings entered from the MONITOR use decimal for relay and sensor node ID numbers (the SYMBOLICS would normally use hex encoding).

5.0 o Verify that all Fault Insertion switches are in the `NORMAL' position.

6.0 o Enter a configuration ('config') string, for example:
c01010201040105010601220131013301350 1

o Observe the ASCII Display Device output message "LOADING CONFIGURATION."

o Verify that the proper relay LEDs are lighted (for this example, relays 1, 2, 4, 5, 6, 22, 31, 33, and 35 should be lighted and other relay LEDs should be extinguished).

o At MONITOR, verify the resulting `config' status message, the string for this example would be as follows:

c0101020103000401050106010700080009001000110012001300140015001600170018001900200021002201230024002500260027002800290030003101320033013400350136003700380039004000410042004300440045004600470048 00

o Verify that Power Supplies 1 and 2 of Power Module 1 are supplying current and that the remaining supplies are not.

7.0 o At this point SDAS.EXE will wait for the bread board to achieve a steady state condition that will be indicated by an ASCII Display Device message "STEADY STATE ACHIEVED".

o Verify that some rendition of the following `steady state' sensor node data string is output at the MONITOR: (Slightly varying conditions will cause some deviation in voltage and current values output in the `steady state' string.)

s017507027407036E0204710E056D10065F10070300080300090000100000
110000120000130300140300150 0FF1600001700FF180000195F232000FF2
100FF226C00236D00246200

8.0 o SDAS.EXE will then enter a "looking for fault" mode that periodically compares new measurement data to the steady state data. (A difference filter of +3 and -3 counts is applied to minimize the effects of spontaneous noise.)

o Insert a fault into the configuration. (For this example, set the RELAY FAULT TYPE switch to OPEN and the fault insertion switch for Relay 1 to RELAY. This fault will cause the current supplied by Power Supply 1 to be interrupted completely thus requiring Power Supply 2 to take on the additional current load.)

9.0 o Upon detection of the fault, "FAULT DETECTED" will be output on the ASCII Display Device.

o Verify that some rendition of the following `fault state' sensor node data string is output at the MONITOR: (Slightly varying conditions will cause some deviation in voltage and current values output in the 'fault state' string.)

f017300002740E036D0104710E056D10066010070300080300090000100000
110000120000130300140300150 0FF1600001700FF180000195F242000FF2
100FF226C00236D00246200

10.0 o Change the rotary switch setting to FAULT NODES

o Observe the ASCII Display Device which should be displaying data for sensor node 1 (N01) in the following form: (Values may not be exactly the same but they should be reasonable.)

N01 00.000a <= 00.780a

The first value represents the fault state node current while the second value represents the steady state node current. As expected, the current for Power Supply should be zero (or slightly positive or negative) after the fault is inserted.

- Depress the REPEAT FUNCTION switch to advance the fault display to the next mismatching voltage/current measurement which in this case should be:

  N02 01.463a <= 00.780a

  Note that the fault current for Power Supply 2 should be close to twice the steady state value.

- Depressing the REPEAT FUNCTION switch again should result in an ASCII Display Device message that reads "FAULT DISPLAY FINISHED". If not, reactivate the switch to scroll through any other voltage/current mismatches that have been detected until the above message is displayed.

11.0 - Return the switch for relay 1 to NORMAL (vs. RELAY), select a NODE I.D. of 01, and set the rotary switch to STATUS.

- The ASCII Display Device will display the present voltage/current measurement data for Sensor Node 1 as follows: (Values may vary.)

  N01 36.563v 00.780a

  The first value represents the node voltage (in volts) and the second value represents the node current (in amperes). As expected the current level supplied by Power Supply 1 has returned to a nominal value.

- Change NODE ID to 02 and depress the REPEAT FUNCTION switch.

- This action will display the voltage/current measurement data for Sensor Node 2 which should be close to:

  N02 36.563v 00.780a

  And again as expected the current supplied by Power Supply 2 has returned to its nominal value.

12.0 - Repeat steps 5.0 through 11.0 for the other configurations and verify that the Bread Board responds properly, that inserted faults can be detected and the voltage/current data is reasonable.

Micro Processor - SYMBOLICS Communication Subsystem Test
(Procedure C)

1.0 o Power on Monitor and SYMBOLICS CRT terminals.

2.0 o Apply power to SYMBOLICS

o To bootstrap SYMBOLICS, enter: logout <return> then: halt machine <return>, watch for the Fep > prompt at the top of the terminal display, and enter b <return>.

o Login to SYMBOLICS by entering:

(login'fies)

o Load and start the SYMBOLICS Interface Software Subsystem:

(load "c: >fies> make-fies-interface.bin")

(make-system'fies-interface':noconfirm)

3.0 o Apply power to the Bread Board (this action will auto-boot the Micro Processor).

o Verify Micro Processor boot and self-test diagnostic results (If problem, refer to INTEL's Software Diagnostic Test Documentation).

4.0 o Rotate Front Panel selector switch to RESET.

o At Monitor CRT terminal, enter:

SDAS.EXE

o After a small delay, the Bread Board should respond with "SYSTEM HAS BEEN RESET" on the ASCII Display Device.

5.0 o At SYMBOLICS terminal, start the test configuration software as follows:

(test-configuration)

o Verify that an unconfigured graphic representation of the FIES network is displayed on the SYMBOLICS terminal.

6.0 o Set all Fault Insertion switches to NORMAL.

o Rotate Fron Panel selector switch to SLAVE MODE.

o Verify ASCII Display Device message reads "SYMBOLICS SRC MODE".

- o If ASCII Display Device message is not as expected, depress the REPEAT FUNCTION and recheck.

7.0
- o At SYMBOLICS terminal, establish a configuration via the mouse input device.
- o Visually reverify the configuration is as desired.
- o Download the configuration via the mouse/menu selection window.
- o Verify that the status message in lower corner of SYMBOLICS terminal reads "XMIT configuration to micro".

8.0
- o Verify that ASCII Display Device message reads "LOADING CONFIGURATION".
- o Verify that the LEDS are lighted for relays that were turned on and extinguished for those relays that were turned off in the configuration download.
- o Verify that the Power Module current meters register a reasonable level of current flow for the selected configuration.
- o Verify that the status message in lower corner of SYMBOLICS terminal reads "IO Wait: Steady State..".

9.0
- o After waiting a few moments, verify that the Bread Board has stabilized by observing the ASCII Display Device message, "STEADY STATE ACHIEVED".
- o Verify that the steady state data stream has been received at the SYMBOLICS by observing "IO Complete: Steady State" in the status display window in the lower right hand corner of the SYMBOLICS terminal.
- o Verify that this status message changes to "IO Wait: Fault Data" which means that the SYMBOLICS is ready for a fault to be inserted.
- o Using the mouse/menu, display the steady state graphic display window and verify that the voltage/current values are reasonable.
- o Return to the previous configuration window and verify that the status is still "IO Wait: Fault Data. (If it is not then spurious noise has caused a false fault report and the test procedure should be restarted from step 6.0.)

10.0 o Insert a fault by depressing one of the Front Panel Fault Insertion switches.

o Verify that the associated LED is extinguished or illuminated if the fault type was an open or closed type fault, respectively.

o Verify that the inserted fault is detected by observing the ASCII Display Device message, "FAULT DETECTED".

o Verify that the fault data string is received at the SYMBOLICS by observing the message in the status window of the SYMBOLICS terminal which should read "IO Complete: Fault Data".

o Using mouse/menu window, display the fault state graphic display window and verify that the voltage/current values are reasonable for the type and location of the fault inserted.

11.0 o Remove the inserted fault by returning the Fault Insertion switch to its NORMAL position.

o If a direct shunt fault was applied, power cycle the associated Power Module to restore the Power Supplies to a "non-folded over" state.

12.0 o Repeat steps 6.0 through 11.0 for all applicable configurations.

SYMBOLICS - FIES Operational Test
(Procedure D)

1.0 o Power on Monitor and Symbolics CRT terminals.

2.0 o Apply power to SYMBOLICS.

o To bootstrap SYMBOLICS, enter: logout <return> then: halt machine <return>, then watch for the Fep > prompt at the top of the terminal display, and enter b <return>.

o Login to SYMBOLICS by entering:

(login'fies:host "d")

3.0 o Apply power to the Bread Board (this action will auto-boot the Micro Pro).

o Verify Micro Pro boot and self-test diagnostic results (If problem, refer to INTEL's Software Diagnostic Test Documentation).

4.0 o Rotate Front Panel selector switch to RESET.

o At Monitor CRT terminal, enter:

SDAS.EXE

o After a small delay, the Bread Board should respond with "SYSTEM HAS BEEN RESET" on the ASCII Display Device.

5.0 o Load and start the ART application software (i.e., the Expert System):

o To enter the ART world enter <select> A at the SYMBOLICS Terminal.

o Using mouse/menu window, clear ART

o Using mouse/menu window, load ART using the following file name:

d: >art-test> load-fies.lisp

- o Using mouse/menu window, reset ART
- o Using mouse/menu window, run ART
- o Verify that an unconfigured graphic representation of the FIES network is displayed on the SYMBOLICS terminal.

6.0
- o Set all Fault Insertion switches to NORMAL.
- o Rotate Front Panel selector switch to SLAVE MODE.
- o Verify ASCII Display Device message reads "SYMBOLICS SRC MODE".
- o If ASCII Display Device message is not as expected, depress the REPEAT FUNCTION and recheck.

7.0
- o At SYMBOLICS terminal, establish a configuration via the mouse input device or file archiving facility.
- o Visually reverify the configuration is as desired.
- o Download the configuration via the mouse/menu selection window.
- o Verify that the status message in lower corner of SYMBOLICS terminal reads "XMIT configuration to micro".

8.0
- o Verify that ASCII Display Device message reads "LOADING CONFIGURATION".
- o Verify that the LEDS are lighted for relays that were turned on and extinguished for those relays that were turned off in the configuration download.
- o Verify that the Power Module current meters register a reasonable level of current flow for the selected configuration.
- o Verify that the status message in lower corner of SYMBOLICS terminal reads "IO Wait: Steady State..".

9.0
- o After waiting a few moments, verify that the Bread Board has stabilized by observing the ASCII Display Device message, "STEADY STATE ACHIEVED".
- o Verify that the steady state data stream has been received at the SYMBOLICS by observing "IO Complete: Steady State" in the status display window in the lower right hand corner of the SYMBOLICS terminal.
- o Verify that this status message changes to "IO Wait: Fault Data" which means that the SYMBOLICS is ready for a fault to be inserted.

- Using the mouse/menu, display the steady state graphic display window and verify that the voltage/current values are reasonable.

- Return to the previous configuration window and verify that the status is still "IO Wait: Fault Data. (if it is not then spurious noise or battery charging has caused a false fault report and the test procedure should be restarted from the 'reset' in step 5.0.)

10.0 - Insert a fault by depressing one of the Front Panel Fault Insertion switches.

- Verify that the associated LED is extinguished.

- Verify that the inserted fault is detected by observing the ASCII Display Device message, "FAULT DETECTED IN SYSTEM".

- Verify that the fault data string is received at the SYMBOLICS by observing the message in the status window of the SYMBOLICS terminal which should read "IO Complete: Fault Data".

11.0 - At SYMBOLICS, wait for completion of the Expert System run.

12.0 - Verify the integrity of the fault hypothesis generated by the Expert System.

13.0 - Remove the inserted fault by returning the Fault Insertion switch to its NORMAL position.

- If a direct shunt fault was applied, power cycle the associated Power Module to restore the Power Supplies to a "non-folded over" state.

14.0 - Repeat step 5.0 (from 'reset') through 13.0 for all applicable configurations.

Formal Acceptance Test Procedure
(Procedure E)

1.0 Overview

This describes the procedure to be used in testing fault isolation capability for any selected case. The procedure is an implementation of the end to end testing specified in the FIES software development plan.

2.0 Case selection

The number of cases have been identified for formal acceptance test. These cases represent the full range of reasonable configurations possible in the power network. A reasonable configuration is selected based on the following criteria:

1) A case must be representative of a possible power system configuration.

2) A case must not cause a power overload in the system, or violate any of the conditions described in chapter 6 of the user's manual.

3.0 Procedure

3.1 Complete initialization and integrity testing as described in test procedures A-D.

3.2 For each test configuration in para. 3.3, perform the following steps

3.2.1 Reset bread board

3.2.2 Reset ART application

3.2.3 Run ART application

3.2.4 Select a configuration from configuration library

3.2.5 Download configuration to the breadboard

3.2.6 When the I/OWait: Fault Data message appears in the FIES Interface window, insert the desired fault into the breadboard

3.2.7 Verify results against expected results. Expected results appear in 3.3 below with the test configuration case descriptions.

3.3 Description

3.3.1 Single Source to single-sink, low load requirements

3.3.1.1 Resistive Short Circuits (Example F-1)

3.3.1.2 Direct Short Circuit (Example F-2)

3.3.1.3 Open Circuits (Example F-3)

3.3.2 Two Sources to multiple sinks, low load requirements

3.3.2.1 Resistive Short Circuits (Example F-4)

3.3.2.2 Direct Short Circuits (Example F-5)

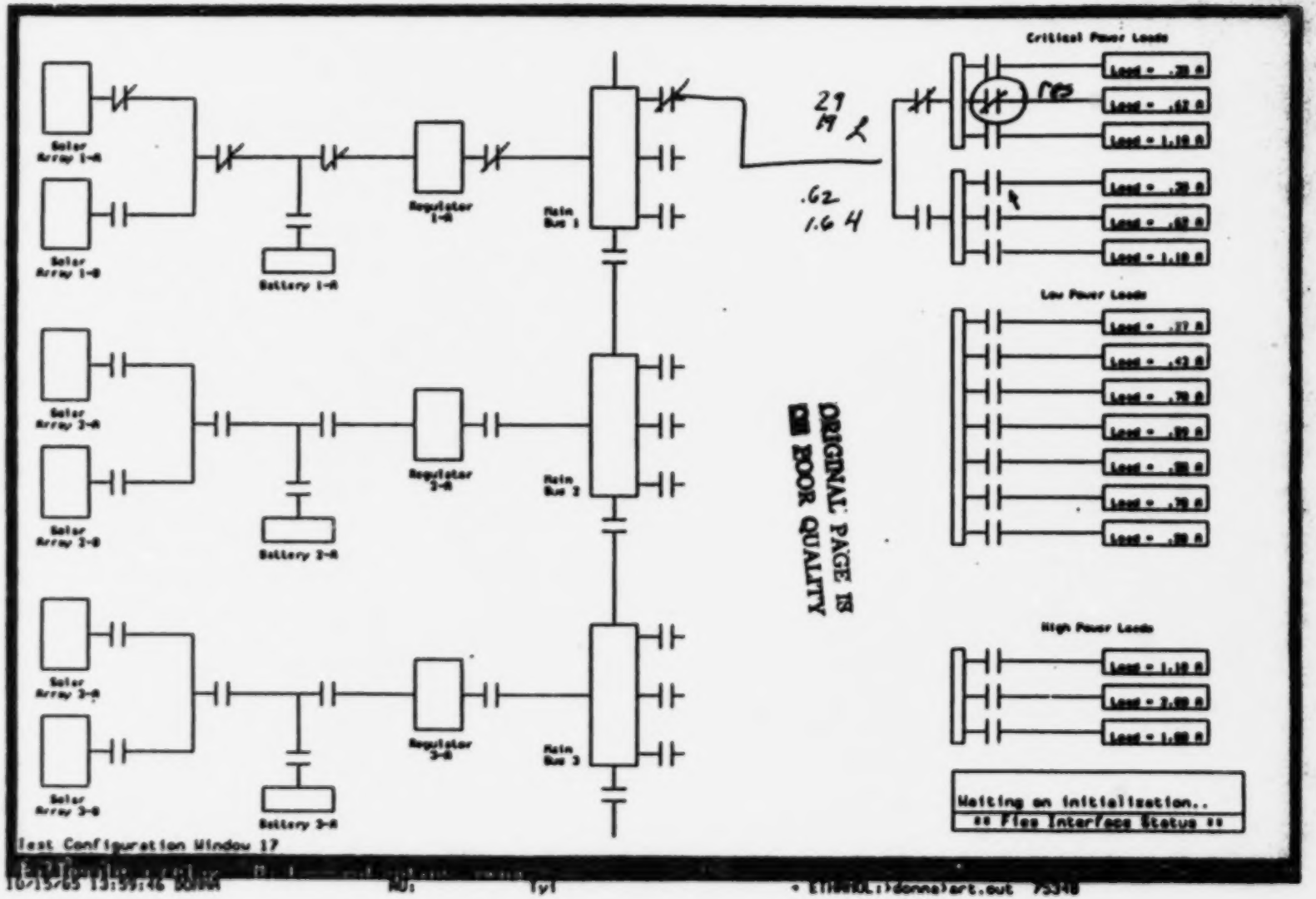
3.3.2.3 Open Circuits (Example F-6)

3.3.3 Four Sources to multiple sinks, high load requirements

3.3.3.1 Resistive Short Circuits (Example F-7)

3.3.3.2 Direct Short Circuits (Example F-8)

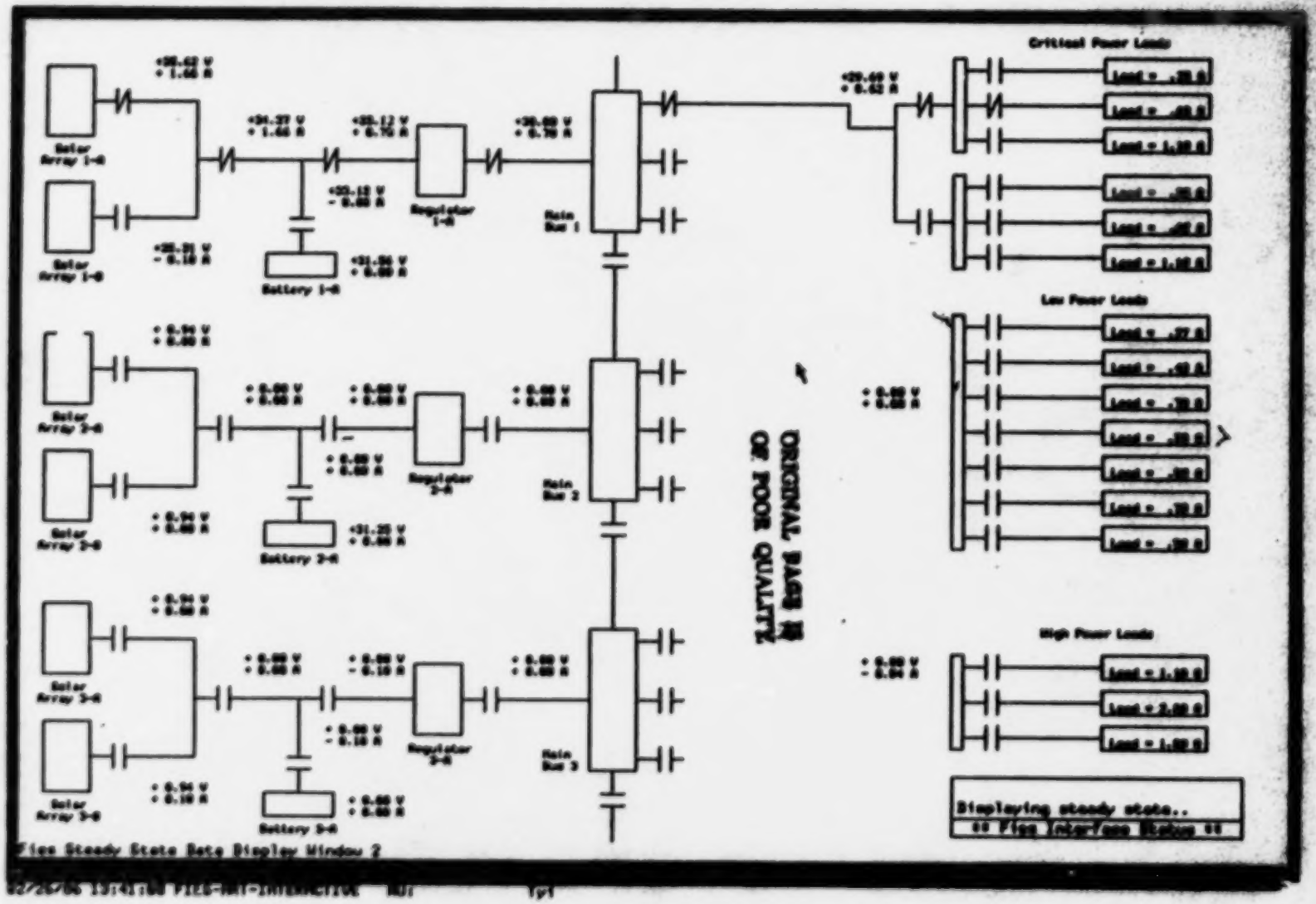
3.3.3.3 Open Circuits (Example F-9)



ORIGINAL PAGE IS
OF POOR QUALITY

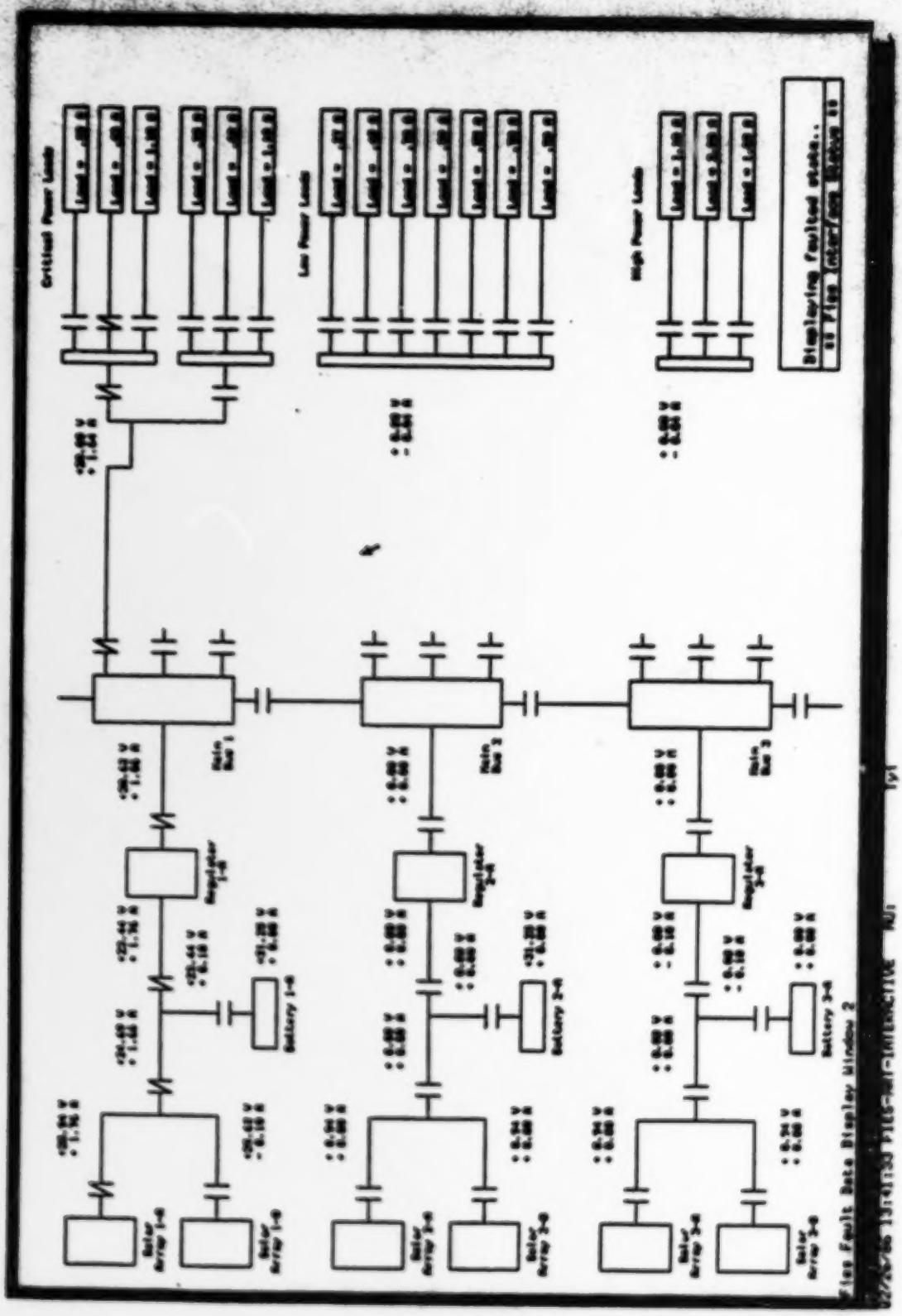
Fig 1

res 45 dat



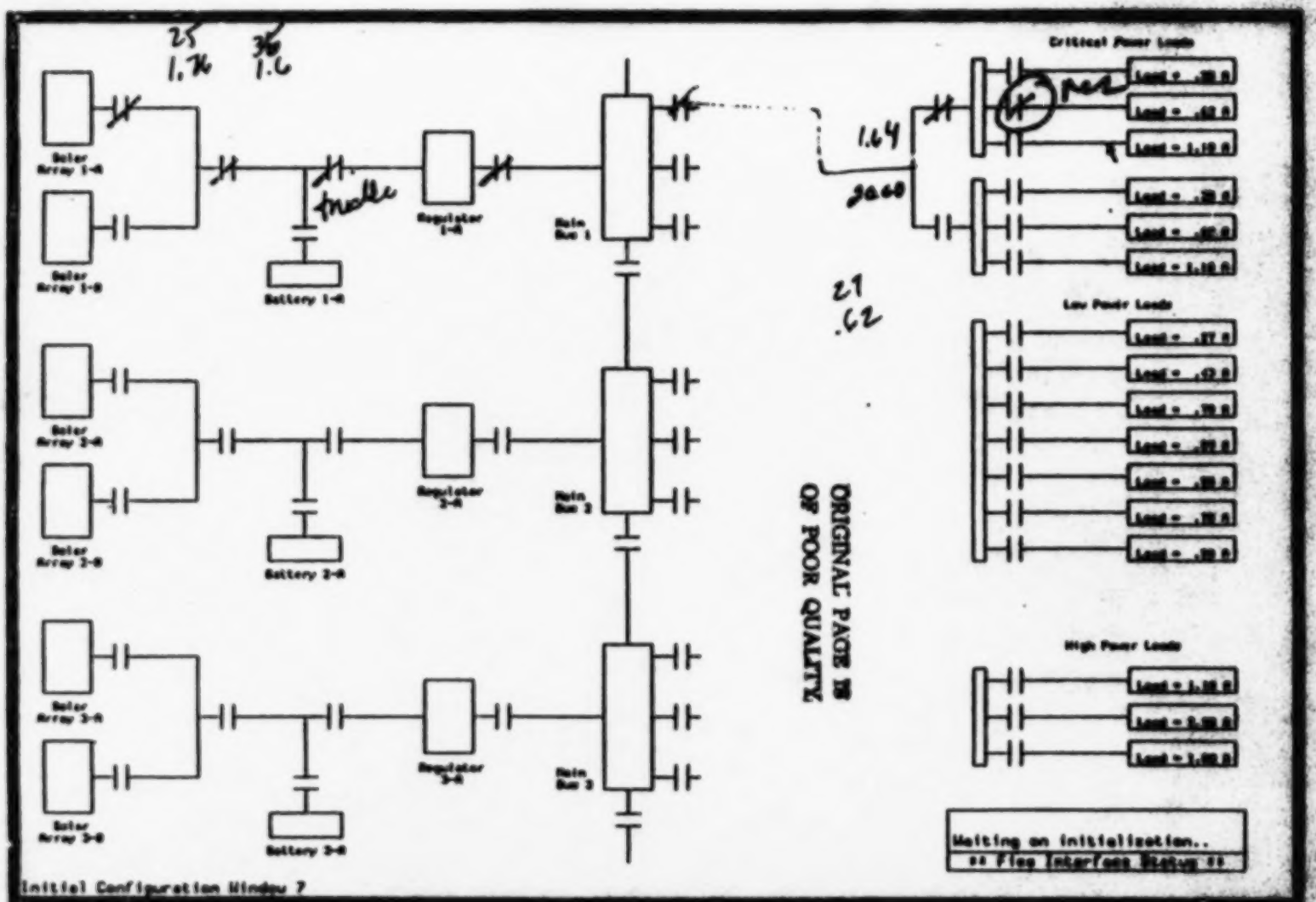
ORIGINAL PAGE IS
OF POOR QUALITY

Fig. A-1



F-1

ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS OF POOR QUALITY

DONNA-1

As expected.

RES45.DAT

```
=> pop
=> run
Beginning run at 2/26/88 13:42:43: type ^C to halt.
 Initializing Circuit Description

 Create Errorbands

 Assigning Qualitative Values to Sensors

 Hypothesizing Faults:

Fault-Hypothesis: Closed-Relay RELAY-1N

Fault-Hypothesis: Closed-Relay RELAY-1P

Fault-Hypothesis: Resistive-Short-Circuit BUS-1D

Fault-Hypothesis: Resistive-Short-Circuit BUS-1E

Fault-Hypothesis: Resistive-Short-Circuit LOAD-1O

Fault-Hypothesis: Short-Circuit BUS-1D

Fault-Hypothesis: Short-Circuit BUS-1E

Fault-Hypothesis: Short-Circuit LOAD-1O

Reject Fault Hypothesis Short-Circuit at LOAD-1O
Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-1G

Reject Fault Hypothesis Short-Circuit at BUS-1E
Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-1G

Reject Fault Hypothesis Short-Circuit at BUS-1D
Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-1G

Reject Fault Hypothesis Closed-Relay at RELAY-1P
Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-1G

Reject Fault Hypothesis Closed-Relay at RELAY-1N
Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-1G

Fault-hyp: Resistive-Short-Circuit BUS-1E
Source Propagation from BUS-1E to RELAY-1K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from RELAY-1K to BUS-1D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from BUS-1D to SENSOR-1G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1G
Source Propagation from SENSOR-1G to RELAY-1G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from RELAY-1G to BUS-1C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from BUS-1C to SENSOR-1F

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1F
Source Propagation from SENSOR-1F to RELAY-1F
```

F-1

6

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from RELAY-1F to REGULATOR-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from REGULATOR-1A to SENSOR-1E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1E
Source Propagation from SENSOR-1E to RELAY-1E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from RELAY-1E to BUS-1B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from BUS-1B to SENSOR-1C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1C
Source Propagation from SENSOR-1C to RELAY-1C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from RELAY-1C to BUS-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from BUS-1A to SENSOR-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1A
Source Propagation from SENSOR-1A to RELAY-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Source Propagation from RELAY-1A to SOLAR-ARRAY-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
End-Source Propagation from SOLAR-ARRAY-1A

Fault-hyp: Resistive-Short-Circuit BUS-1E
Sink Propagation from BUS-1E to RELAY-1D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
Sink Propagation from RELAY-1D to LOAD-1D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1E
End-Sink Propagation from LOAD-1D

Fault-hyp: Resistive-Short-Circuit BUS-1D
Source Propagation from BUS-1D to SENSOR-1G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1G
Source Propagation from SENSOR-1G to RELAY-1G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Source Propagation from RELAY-1G to BUS-1C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Source Propagation from BUS-1C to SENSOR-1F

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1F
Source Propagation from SENSOR-1F to RELAY-1F

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Source Propagation from RELAY-1F to REGULATOR-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Source Propagation from REGULATOR-1A to SENSOR-1E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Predicted Current Effect: HIGH Consistent at Sensor : NSOR-1E
Source Pro agation from SENSOR-1E to RELAY-1E

-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
: Propagation from RELAY-1E to BUS-1B

Ex. B-1

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Source Propagation from BUS-1B to SENSOR-1C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1C
Source Propagation from SENSOR-1C to RELAY-1C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Source Propagation from RELAY-1C to BUS-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Source Propagation from BUS-1A to SENSOR-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1A
Source Propagation from SENSOR-1A to RELAY-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Source Propagation from RELAY-1A to SOLAR-ARRAY-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
End-Source Propagation from SOLAR-ARRAY-1A

Fault-hyp: Resistive-Short-Circuit BUS-1D
Sink Propagation from BUS-1D to RELAY-1K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Sink Propagation from RELAY-1K to BUS-1E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Sink Propagation from BUS-1E to RELAY-1O

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
Sink Propagation from RELAY-1O to LOAD-1O

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-1D
End-Sink Propagation from LOAD-1O

Fault-hyp: Resistive-Short-Circuit LOAD-1O
Source Propagation from LOAD-1O to RELAY-1O

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Source Propagation from RELAY-1O to BUS-1E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Source Propagation from BUS-1E to RELAY-1K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Source Propagation from RELAY-1K to BUS-1D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Source Propagation from BUS-1D to SENSOR-1G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1G
Source Propagation from SENSOR-1G to RELAY-1G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Source Propagation from RELAY-1G to BUS-1C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Source Propagation from BUS-1C to SENSOR-1F

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1F
Source Propagation from SENSOR-1F to RELAY-1F

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Source Propagation from RELAY-1F to REGULATOR-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
Source Propagation from REGULATOR-1A to SENSOR-1E

-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-1O
cted Current Effect: HIGH Consistent at Sensor SENSOR-1E
ce Propagation from SENSOR-1E to RELAY-1E

Ex. F-1

7

```
Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-10
Source Propagation from RELAY-1E to BUS-1B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-10
Source Propagation from BUS-1B to SENSOR-1C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-10
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1C
Source Propagation from SENSOR-1C to RELAY-1C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-10
Source Propagation from RELAY-1C to BUS-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-10
Source Propagation from BUS-1A to SENSOR-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-10
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-1A
Source Propagation from SENSOR-1A to RELAY-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-10
Source Propagation from RELAY-1A to SOLAR-ARRAY-1A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-10
End-Source Propagation from SOLAR-ARRAY-1A

Reporting All Initial Fault Hypotheses:

     SHORT-CIRCUIT LOAD-10

     SHORT-CIRCUIT BUS-1E

     SHORT-CIRCUIT BUS-1D

     RESISTIVE-SHORT-CIRCUIT LOAD-10

     RESISTIVE-SHORT-CIRCUIT BUS-1E

     RESISTIVE-SHORT-CIRCUIT BUS-1D

     CLOSED-RELAY RELAY-1P

     CLOSED-RELAY RELAY-1N

Reporting Final Fault Hypotheses:

      RESISTIVE-SHORT-CIRCUIT LOAD-10

      RESISTIVE-SHORT-CIRCUIT BUS-1E

      RESISTIVE-SHORT-CIRCUIT BUS-1D

Readings at SENSOR-3Z
  Normal  Voltage:  0.0
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL

Readings at SENSOR-2Z
  Normal  Voltage:  31.24992
  Faulted Voltage:  31.24992
  Qual Voltage:     NORMAL

     mal Current:   0.0
     lted Current:  0.0
```

Ex: F-1

```
  Qual Current:     NORMAL


Readings at SENSOR-1Z
  Normal  Voltage:  31.562418
  Faulted Voltage:  31.24992
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-3G
  Normal  Voltage:  0.0
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   -0.0390624
  Faulted Current:  -0.0390624
  Qual Current:     NORMAL


Readings at SENSOR-2G
  Normal  Voltage:  0.0
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  -0.0390624
  Qual Current:     NORMAL


Readings at SENSOR-1G
  Normal  Voltage:  29.887424
  Faulted Voltage:  19.999949
  Qual Voltage:     LOW

  Normal Current:   0.6249984
  Faulted Current:  1.6406208
  Qual Current:     HIGH


Readings at SENSOR-3F
  Normal  Voltage:  0.0
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-3E
  Normal  Voltage:  0.0
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   -0.097656
  Faulted Current:  -0.097656
  Qual Current:     NORMAL
```

F-1
9

Readings at SENSOR-3C
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3D
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: -0.097856
Faulted Current: -0.097856
Qual Current: NORMAL

Readings at SENSOR-3B
Normal Voltage: 0.9374876
Faulted Voltage: 0.9374876
Qual Voltage: NORMAL

Normal Current: 0.097856
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3A
Normal Voltage: 0.9374876
Faulted Voltage: 0.9374876
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2F
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2E
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2C
Normal Voltage: 0.0
Faulted Voltage: 0.0

Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2D
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2B
Normal Voltage: 0.9374876
Faulted Voltage: 0.9374876
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2A
Normal Voltage: 0.9374876
Faulted Voltage: 0.9374876
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-1F
Normal Voltage: 29.[illegible]924
Faulted Voltage: 20.624947
Qual Voltage: LOW

Normal Current: 0.781248
Faulted Current: 1.[illegible]464
Qual Current: HIGH

Readings at SENSOR-1E
Normal Voltage: 33.124816
Faulted Voltage: 23.437438
Qual Voltage: LOW

Normal Current: 0.781248
Faulted Current: 1.757808
Qual Current: HIGH

Readings at SENSOR-1C
Normal Voltage: 34.374912
Faulted Voltage: 24.687437
Qual Voltage: LOW

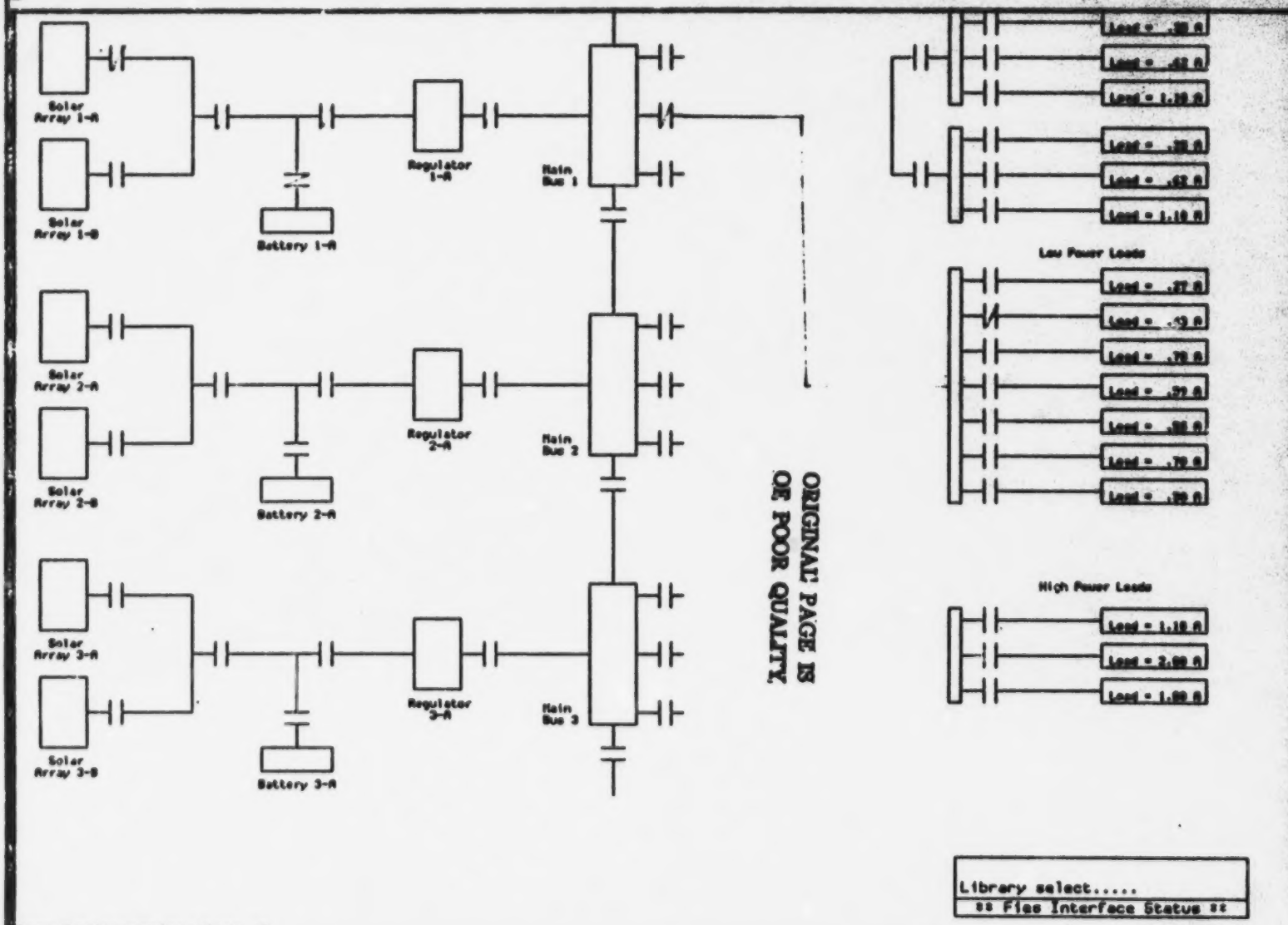
[illegible]al Current: 1.880152
[illegible]ted Current: 1.880152

F-1

11

```
  Qual Current:      NORMAL


Readings at SENSOR-1D
  Normal  Voltage:  33.124816
  Faulted Voltage:  23.437430
  Qual Voltage:      LOW

  Normal Current:   -0.878004
  Faulted Current:  0.087806
  Qual Current:      NORMAL


Readings at SENSOR-1B
  Normal  Voltage:  35.31241
  Faulted Voltage:  25.624833
  Qual Voltage:      LOW

  Normal Current:   -0.087856
  Faulted Current:  -0.087856
  Qual Current:      NORMAL


Readings at SENSOR-1A
  Normal  Voltage:  35.62481
  Faulted Voltage:  25.937433
  Qual Voltage:      LOW

  Normal Current:   1.680152
  Faulted Current:  1.757808
  Qual Current:      NORMAL


Program halted.
Ending run at 2/26/86 13:43:48.
 => watch
 => dribble
```

Please enter the path and filename. Default is d:>files>files-library>

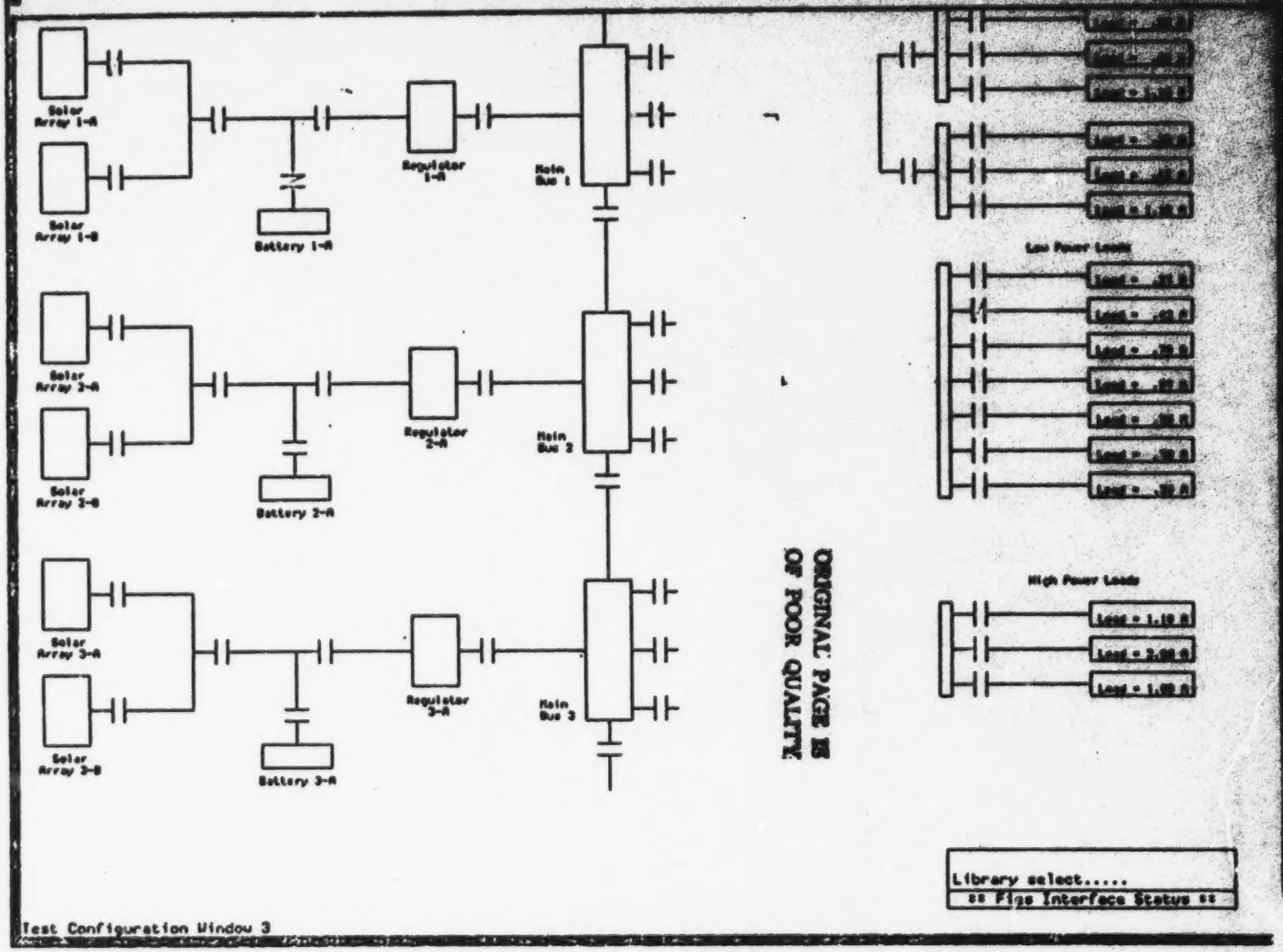


Test Configuration Window 3

ORIGINAL PAGE IS OF POOR QUALITY

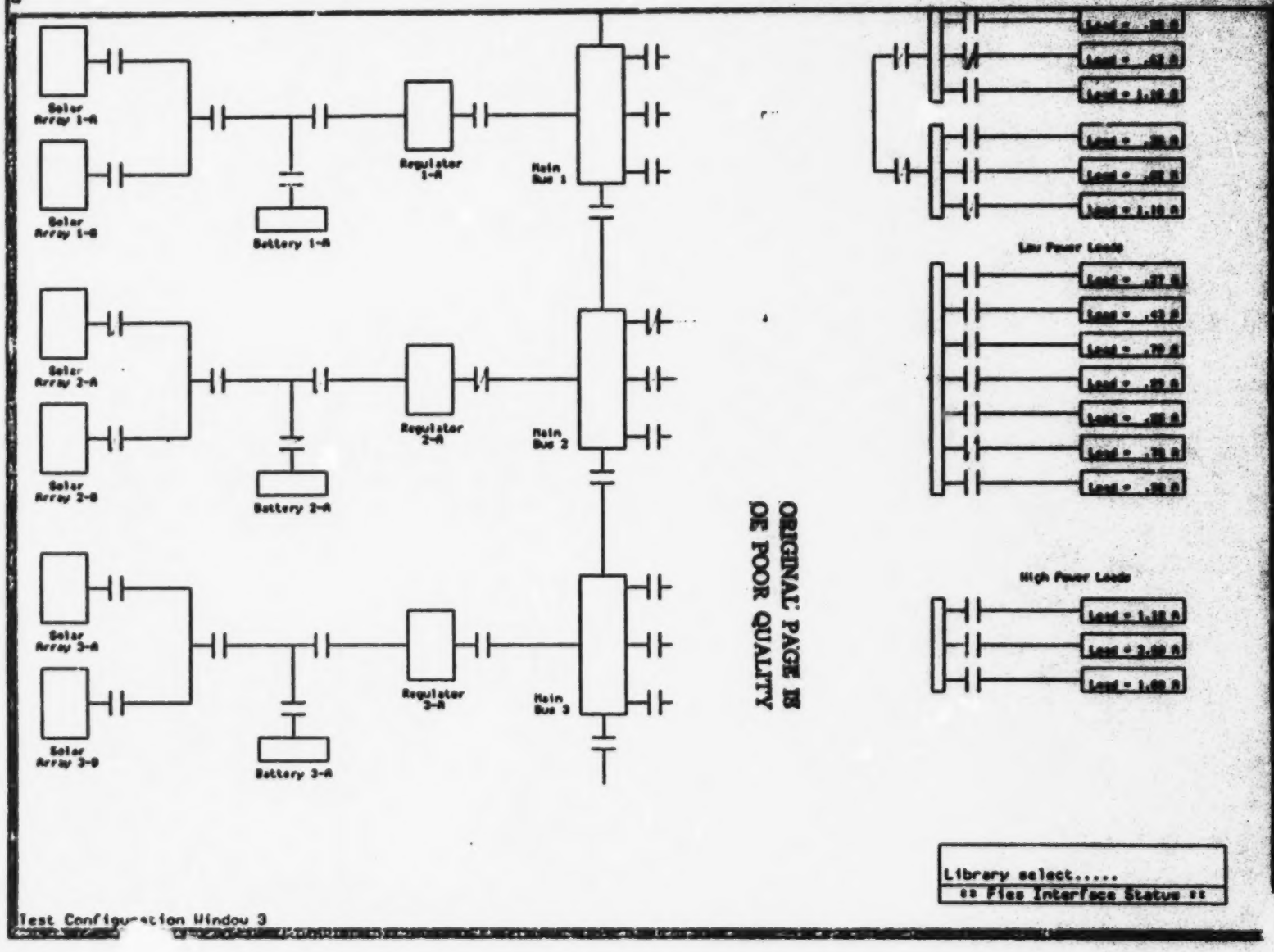
Fig. F-2

Please enter the path and filename. Default is d:>files>fies-library>



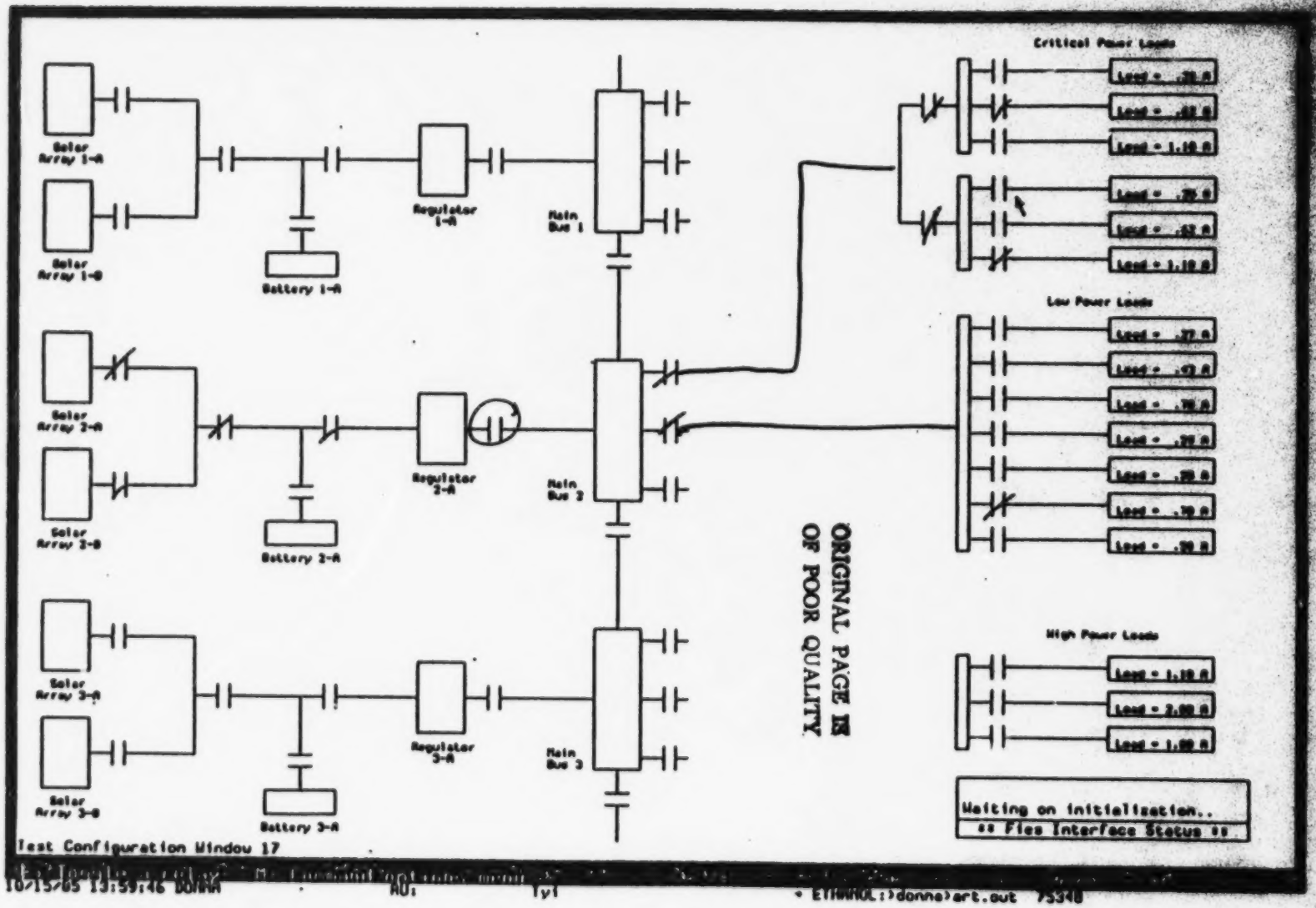
Test Configuration Window 3

ORIGINAL PAGE IS OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

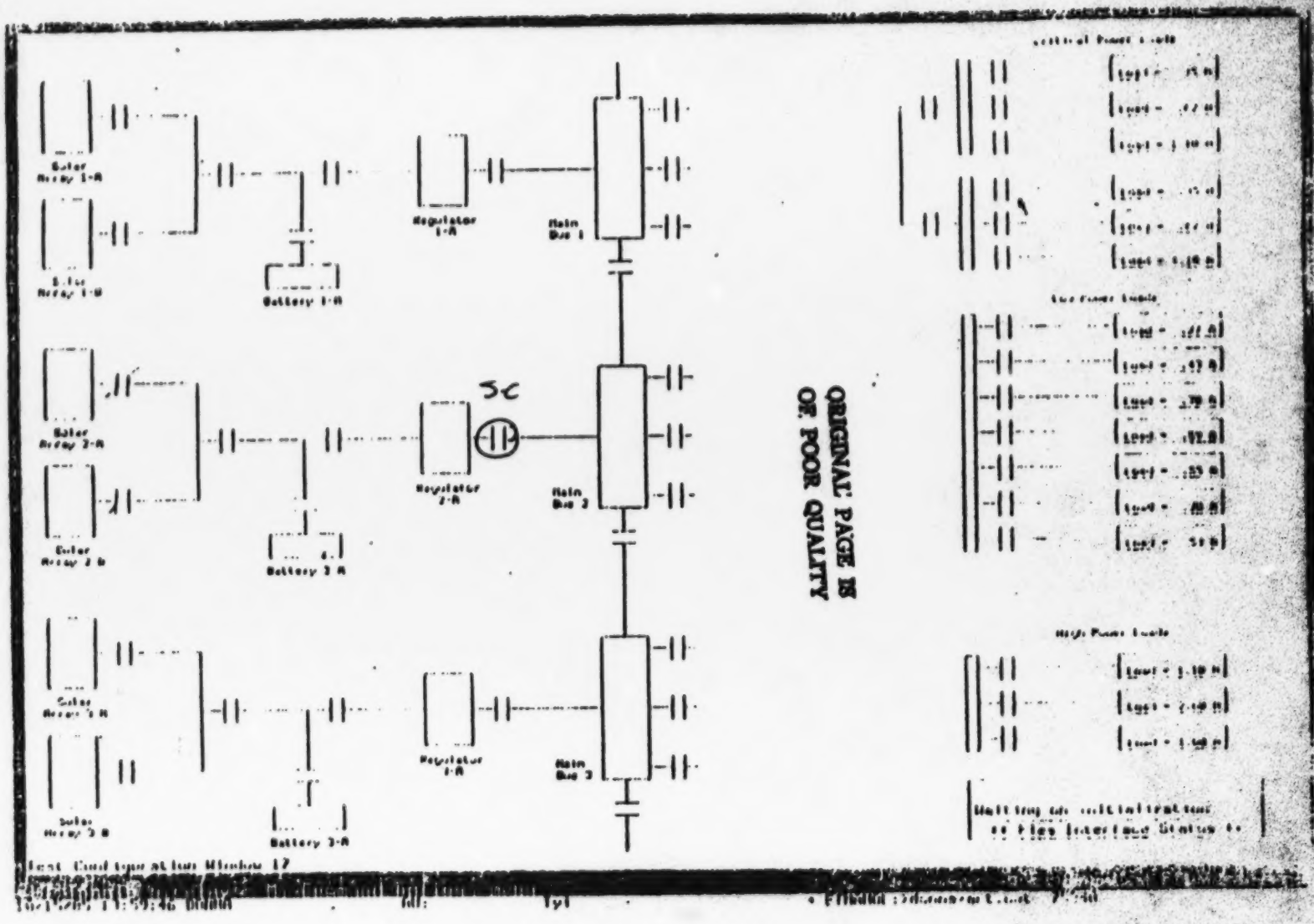
E-4



ORIGINAL PAGE IS
OF POOR QUALITY

Fig: F-5

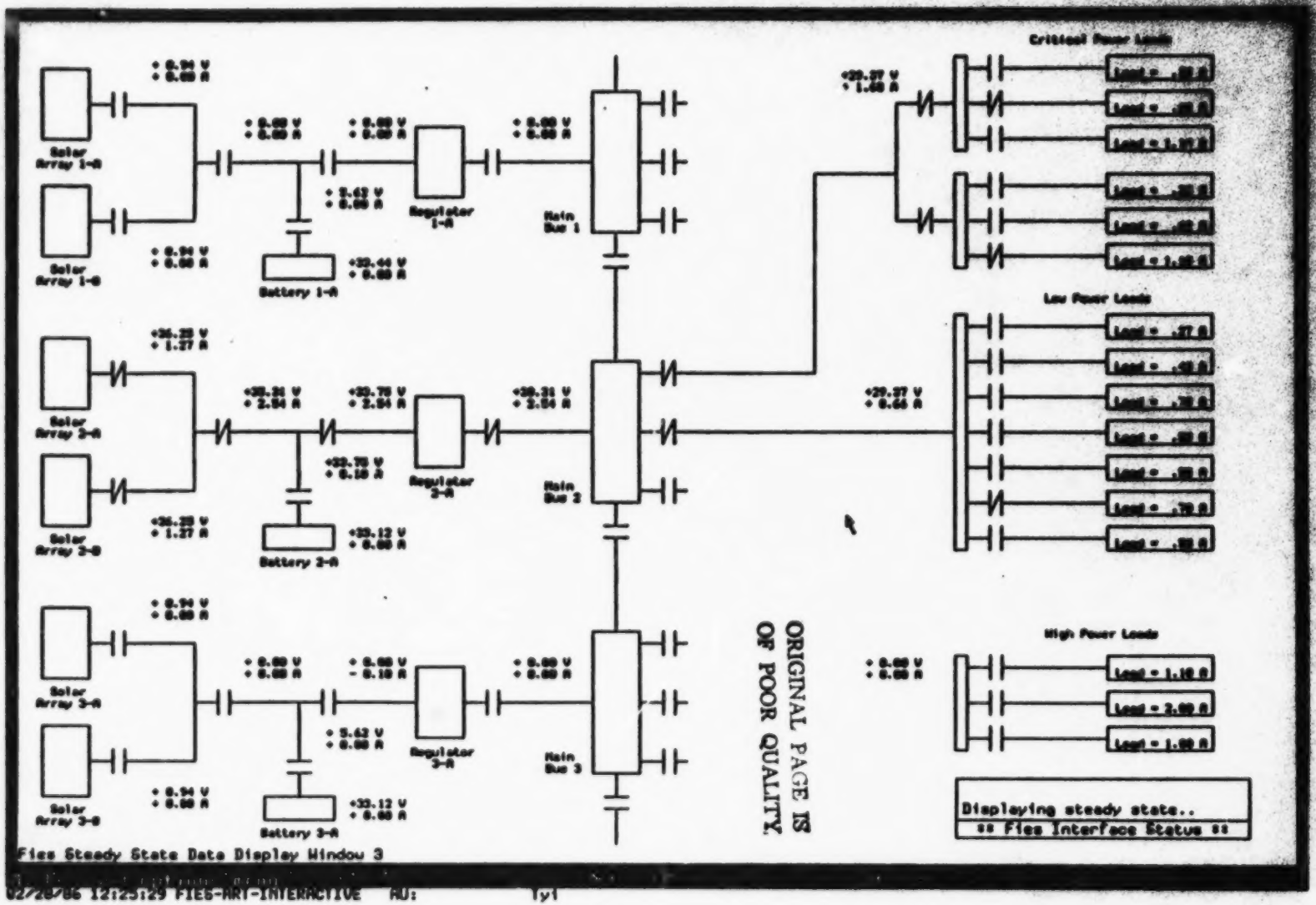
short D.



ORIGINAL PAGE IS
OF POOR QUALITY

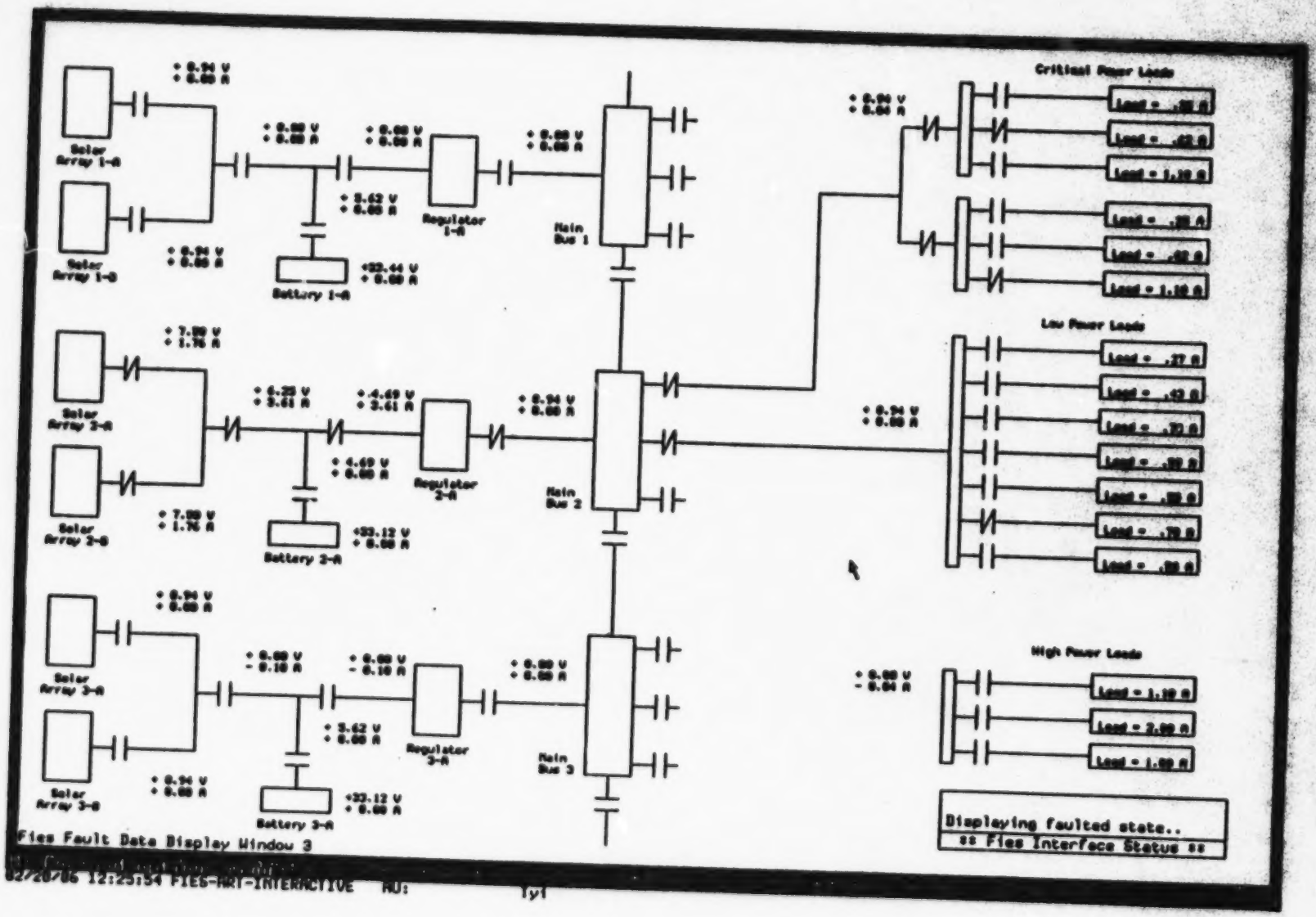
SHORTI[illegible]. DAT

Fig. F-5



ORIGINAL PAGE IS OF POOR QUALITY

Fig 5


Critical Power Loads
Low Power Loads
High Power Loads
Solar Array 1-A
Solar Array 1-B
Solar Array 2-A
Solar Array 2-B
Solar Array 3-A
Solar Array 3-B
Battery 1-A
Battery 2-A
Battery 3-A
Regulator 1-A
Regulator 2-A
Regulator 3-A
Main Bus 1
Main Bus 2
Main Bus 3
Displaying faulted state..
** Fies Interface Status **
Fies Fault Data Display Window 3
02/20/86 12:25:54 FIES-ART-INTERACTIVE

F-5

4

```
=> pop
=> run
Beginning run at 2/28/86 12:27:05: type ^C to halt.
 Initializing Circuit Description


 Create Errorbands


 Assigning Qualitative Values to Sensors


 Hypothesizing Faults:


Fault-Hypothesis: Short-Circuit REGULATOR-2A


Fault-hyp: Short-Circuit REGULATOR-2A
Source Propagation from REGULATOR-2A to SENSOR-2E

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2E
Source Propagation from SENSOR-2E to RELAY-2E

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Source Propagation from RELAY-2E to BUS-2B

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Source Propagation from BUS-2B to SENSOR-2C

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2C
Source Propagation from SENSOR-2C to RELAY-2C

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: Short-Circuit REGULATOR-2A
Sink Propagation from REGULATOR-2A to RELAY-2F

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from RELAY-2F to SENSOR-2F


Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2F
Sink Propagation from SENSOR-2F to BUS-2C

      hyp: SHORT-CIRCUIT REGULATOR-2A
```

Ex: E-5
5

Sink Propagation from BUS-2C to RELAY-2G

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from RELAY-2G to SENSOR-1G

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-1G
Sink Propagation from SENSOR-1G to BUS-1D

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from BUS-1D to RELAY-1K

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from RELAY-1K to BUS-1E

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from BUS-1E to RELAY-1O

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from RELAY-1O to LOAD-1O

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
End-Sink Propagation from LOAD-1O

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from BUS-1D to RELAY-1M

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from RELAY-1M to BUS-1F

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from BUS-1F to RELAY-1S

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from RELAY-1S to LOAD-1S

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
End-Sink Propagation from LOAD-1S

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from BUS-2C to RELAY-2H

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from RELAY-2H to SENSOR-2G

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from BUS-2D to RELAY-2Q

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: SHORT-CIRCUIT REGULATOR-2A
End-Sink Propagation from LOAD-2Q

Reporting All Initial Fault Hypotheses:

SHORT-CIRCUIT REGULATOR-2A

Reporting Final Fault Hypotheses:

SHORT-CIRCUIT REGULATOR-2A

Readings at SENSOR-3Z
[illegible]l Voltage: 33.124916
[illegible]ed Voltage: 33.124916
Voltage: NORMAL

Fig: F-5

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2Z
Normal Voltage: 33.124816
Faulted Voltage: 33.124816
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-1Z
Normal Voltage: 33.437412
Faulted Voltage: 33.437412
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3G
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: -0.0390624
Qual Current: NORMAL

Readings at SENSOR-2G
Normal Voltage: 29.374924
Faulted Voltage: 0.9374976
Qual Voltage: ZERO

Normal Current: 0.6640605
Faulted Current: 0.0
Qual Current: ZERO

Readings at SENSOR-1G
Normal Voltage: 29.374924
Faulted Voltage: 0.9374976
Qual Voltage: ZERO

Normal Current: 1.6796832
Faulted Current: 0.0390624
Qual Current: ZERO

Readings at SENSOR-3F
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

F-5

Readings at SENSOR-3E
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: -0.097656
Faulted Current: -0.097656
Qual Current: NORMAL

Readings at SENSOR-3C
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: -0.097656
Qual Current: NORMAL

Readings at SENSOR-3D
Normal Voltage: 5.6249857
Faulted Voltage: 5.6249857
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3B
Normal Voltage: 0.8374976
Faulted Voltage: 0.8374976
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3A
Normal Voltage: 0.8374976
Faulted Voltage: 0.8374976
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2F
Normal Voltage: 30.312422
Faulted Voltage: 0.8374976
Qual Voltage: ZERO

Normal Current: 2.5390558
Faulted Current: 0.0
Qual Current: ZERO

Ex. F-5

Readings at SENSOR-2E
Normal Voltage: 33.749912
Faulted Voltage: 4.687488
Qual Voltage: LOW

Normal Current: 2.5390558
Faulted Current: 3.613272
Qual Current: HIGH

Readings at SENSOR-2C
Normal Voltage: 36.31241
Faulted Voltage: 6.249984
Qual Voltage: LOW

Normal Current: 2.5390558
Faulted Current: 3.613272
Qual Current: HIGH

Readings at SENSOR-2D
Normal Voltage: 33.749912
Faulted Voltage: 4.687488
Qual Voltage: LOW

Normal Current: 0.097656
Faulted Current: 0.0
Qual Current: ZERO

Readings at SENSOR-2B
Normal Voltage: 38.24991
Faulted Voltage: 7.499981
Qual Voltage: LOW

Normal Current: 1.2695279
Faulted Current: 1.757808
Qual Current: HIGH

Readings at SENSOR-2A
Normal Voltage: 38.24991
Faulted Voltage: 7.499981
Qual Voltage: LOW

Normal Current: 1.2695279
Faulted Current: 1.757808
Qual Current: HIGH

Readings at SENSOR-1F
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-1E
Normal Voltage: 0.0
[illegible]ted Voltage: 0.0
[illegible] Voltage: NORMAL

F-5
9

```
  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-1C
  Normal  Voltage:  0.0
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-1D
  Normal  Voltage:  5.8249857
  Faulted Voltage:  5.8249857
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-1B
  Normal  Voltage:  0.9374976
  Faulted Voltage:  0.9374976
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-1A
  Normal  Voltage:  0.9374976
  Faulted Voltage:  0.9374976
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Program halted.
Ending run at 2/28/86 12:28:07.
 => watch
 => dribble
```

Ex: F5

Example F-6

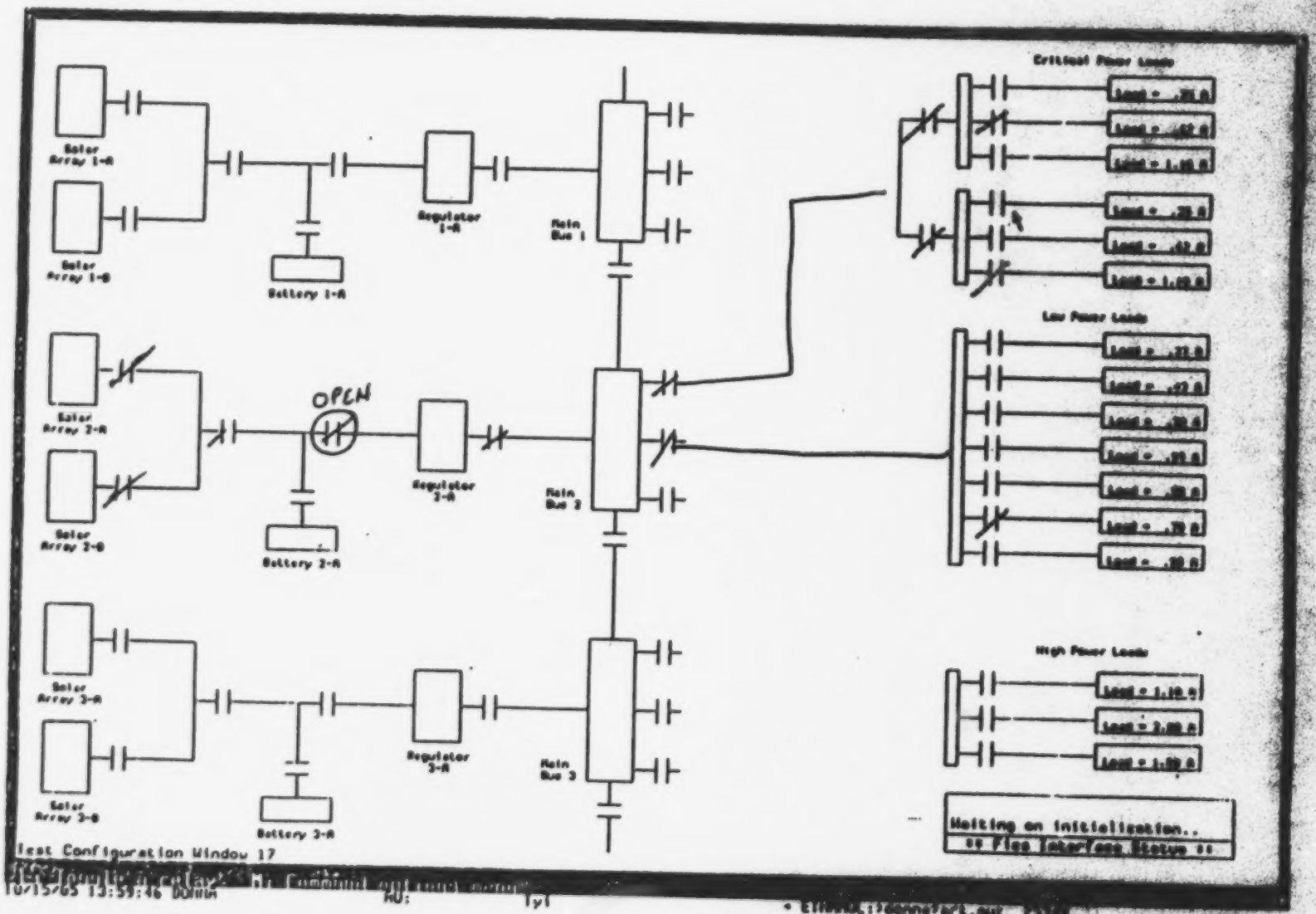
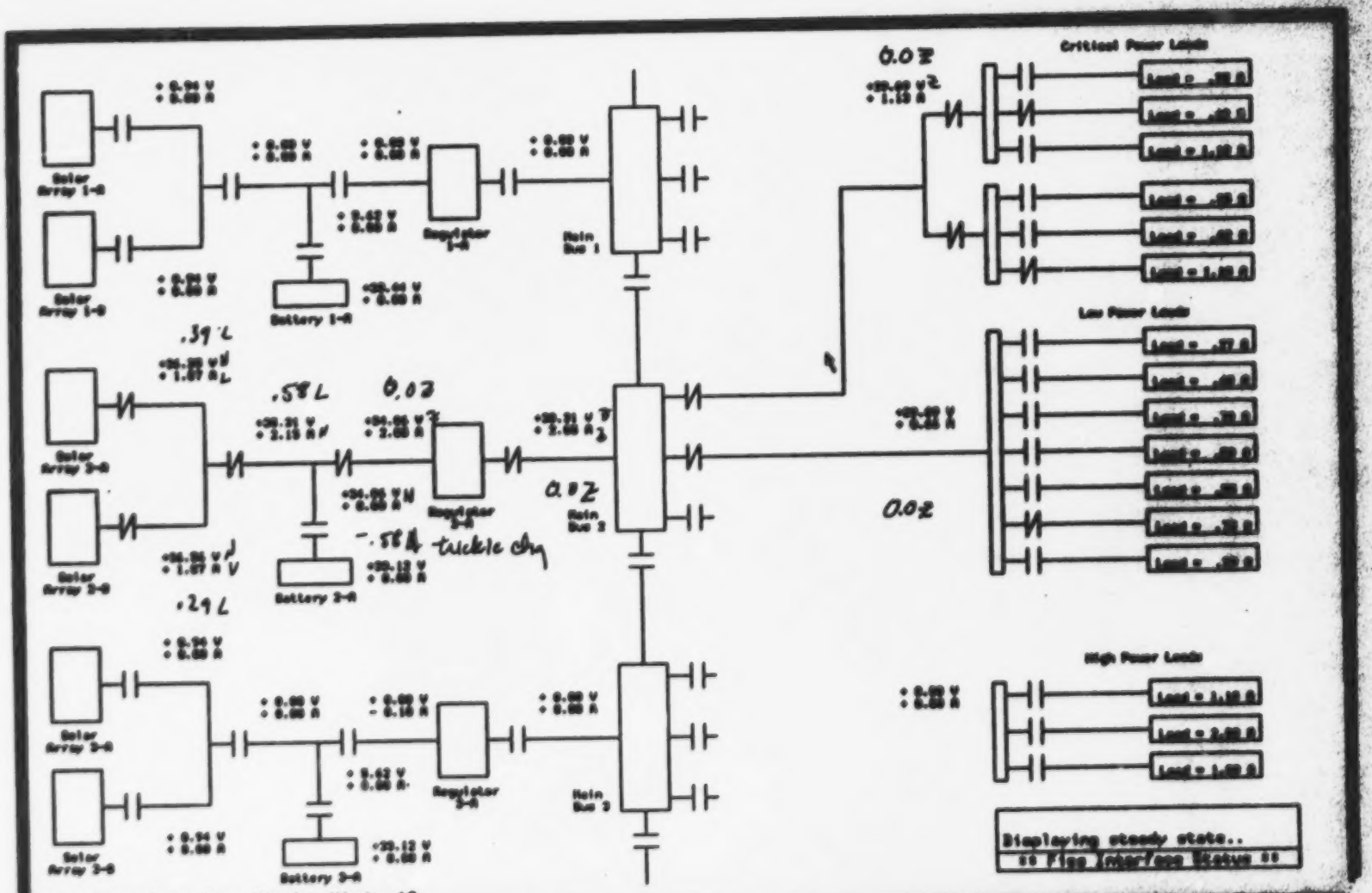


Fig. F-6

ORIGINAL PAGE IS
OF POOR QUALITY

Modified


Critical Power Loads
Low Power Loads
High Power Loads
Solar Array 1-A
Solar Array 1-B
Solar Array 2-A
Solar Array 2-B
Solar Array 3-A
Solar Array 3-B
Battery 1-A
Battery 2-A
Battery 3-A
Regulator 1-A
Regulator 2-A
Regulator 3-A
Main Bus 1
Main Bus 2
Main Bus 3
.39 L
.58 L
0.0 Z
.29 L
-.58 A trickle chg
Displaying steady state..
** Fies Interface Status **
Fies Steady State Data Display Window 13
FIES-MAT-INTERACTIVE

open 7

Ex: P-6
2

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: OPEN-CIRCUIT RELAY-2E
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: OPEN-CIRCUIT RELAY-2E
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: Open-Circuit RELAY-2E
Sink Propagation from RELAY-2E to SENSOR-2E

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2E
Sink Propagation from SENSOR-2E to REGULATOR-2A

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from REGULATOR-2A to RELAY-2F

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from RELAY-2F to SENSOR-2F

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2F
Sink Propagation from SENSOR-2F to BUS-2C

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from BUS-2C to RELAY-2G

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from RELAY-2G to SENSOR-1G

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-1G
Sink Propagation from SENSOR-1G to BUS-1D

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from BUS-1D to RELAY-1K

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from RELAY-1K to BUS-1E

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from BUS-1E to RELAY-1O

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from RELAY-1O to LOAD-1O

Fault-hyp: OPEN-CIRCUIT RELAY-2E
End-Sink Propagation from LOAD-1O

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from BUS-1D to RELAY-1H

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from RELAY-1H to BUS-1F

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from BUS-1F to RELAY-1S

Fig: F-6

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from RELAY-1S to LOAD-1S

Fault-hyp: OPEN-CIRCUIT RELAY-2E
End-Sink Propagation from LOAD-1S

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from BUS-2C to RELAY-2H

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from RELAY-2H to SENSOR-2G

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from BUS-2D to RELAY-2Q

Fault-hyp: OPEN-CIRCUIT RELAY-2E
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: OPEN-CIRCUIT RELAY-2E
End-Sink Propagation from LOAD-2Q

Fault-hyp: Open-Circuit BUS-2B
Source Propagation from BUS-2B to SENSOR-2C

Fault-hyp: OPEN-CIRCUIT BUS-2B
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2C
Source Propagation from SENSOR-2C to RELAY-2C

Fault-hyp: OPEN-CIRCUIT BUS-2B
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: OPEN-CIRCUIT BUS-2B
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: OPEN-CIRCUIT BUS-2B
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: OPEN-CIRCUIT BUS-2B
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: OPEN-CIRCUIT BUS-2B
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: OPEN-CIRCUIT BUS-2B
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: OPEN-CIRCUIT BUS-2B
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: OPEN-CIRCUIT BUS-2B
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: OPEN-CIRCUIT BUS-2B
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: Open-Circuit BUS-2B
Sink Propagation from BUS-2B to RELAY-2E

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from RELAY-2E to SENSOR-2E

Fault-hyp: OPEN-CIRCUIT BUS-2B
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2E
Sink Propagation from SENSOR-2E to REGULATOR-2A

F-6

4

hyp: OPEN-CIRCUIT BUS-2B
Propagation from REGULATOR-2A to RELAY-2F

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from RELAY-2F to SENSOR-2F

Fault-hyp: OPEN-CIRCUIT BUS-2B
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2F
Sink Propagation from SENSOR-2F to BUS-2C

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from BUS-2C to RELAY-2G

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from RELAY-2G to SENSOR-1G

Fault-hyp: OPEN-CIRCUIT BUS-2B
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-1G
Sink Propagation from SENSOR-1G to BUS-1D

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from BUS-1D to RELAY-1K

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from RELAY-1K to BUS-1E

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from BUS-1E to RELAY-1O

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from RELAY-1O to LOAD-1O

Fault-hyp: OPEN-CIRCUIT BUS-2B
End-Sink Propagation from LOAD-1O

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from BUS-1D to RELAY-1M

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from RELAY-1M to BUS-1F

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from BUS-1F to RELAY-1S

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from RELAY-1S to LOAD-1S

Fault-hyp: OPEN-CIRCUIT BUS-2B
End-Sink Propagation from LOAD-1S

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from BUS-2C to RELAY-2H

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from RELAY-2H to SENSOR-2G

Fault-hyp: OPEN-CIRCUIT BUS-2B
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from BUS-2D to RELAY-2Q

Fault-hyp: OPEN-CIRCUIT BUS-2B
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: OPEN-CIRCUIT BUS-2B
End-Sink Propagation from LOAD-2Q

Fault-hyp: Resistive-Short-Circuit SOLAR-ARRAY-2A
Sink Propagation from SOLAR-ARRAY-2A to RELAY-2A

[illegible]-hyp: RESISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2A
[illegible] Propagation from RELAY-2A to SENSOR-2A

[illegible]: F-6

5

Fault-hyp: RESISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2A
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2A
Sink Propagation from SENSOR-2A to BUS-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2A
Sink Propagation from BUS-2A to RELAY-2C
Source Propagation from BUS-2A to SENSOR-2B

Reject Fault Hypothesis RESISTIVE-SHORT-CIRCUIT at SOLAR-ARRAY-2A
Reason: Predicted Effect Is Inconsistent at Sensor SENSOR-2B
Predicted Value: HIGH
Sensor Value: LOW

Fault-hyp: Resistive-Short-Circuit SOLAR-ARRAY-2B
Sink Propagation from SOLAR-ARRAY-2B to RELAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2B
Sink Propagation from RELAY-2B to SENSOR-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2B
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2B
Sink Propagation from SENSOR-2B to BUS-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2B
Sink Propagation from BUS-2A to RELAY-2C
Source Propagation from BUS-2A to SENSOR-2A

Reject Fault Hypothesis RESISTIVE-SHORT-CIRCUIT at SOLAR-ARRAY-2B
Reason: Predicted Effect Is Inconsistent at Sensor SENSOR-2A
Predicted Value: HIGH
Sensor Value: LOW

Reporting All Initial Fault Hypotheses:

RESISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2A

RESISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2B

OPEN-CIRCUIT RELAY-2E

OPEN-CIRCUIT BUS-2B

Reporting Final Fault Hypotheses:

OPEN-CIRCUIT RELAY-2E

OPEN-CIRCUIT BUS-2B

Readings at SENSOR-3Z
Normal Voltage: 33.124916
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2Z
Normal Voltage: 33.124916
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

F-6
6

Readings at SENSOR-1Z
Normal Voltage: 33.437412
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3G
Normal Voltage: 0.0
Faulted Voltage: -0.0390624
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: -0.0390624
Qual Current: NORMAL

Readings at SENSOR-2G
Normal Voltage: 29.687424
Faulted Voltage: 0.0
Qual Voltage: ZERO

Normal Current: 0.6640608
Faulted Current: 0.0
Qual Current: ZERO

Readings at SENSOR-1G
Normal Voltage: 29.687424
Faulted Voltage: 0.0
Qual Voltage: ZERO

Normal Current: 1.1328096
Faulted Current: 0.0
Qual Current: ZERO

Readings at SENSOR-3F
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3E
Normal Voltage: 0.0
Faulted Voltage: -0.097656
Qual Voltage: NORMAL

Normal Current: -0.097656
Faulted Current: -0.097656
Qual Current: NORMAL

Ex: F-6
7

ngs at SENSOR-3C
mal Voltage: 0.0

Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3D
Normal Voltage: 5.6249857
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3B
Normal Voltage: 0.9374976
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3A
Normal Voltage: 0.9374976
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2F
Normal Voltage: 30.312422
Faulted Voltage: 0.0
Qual Voltage: ZERO

Normal Current: 2.050776
Faulted Current: 0.0
Qual Current: ZERO

Readings at SENSOR-2E
Normal Voltage: 34.062412
Faulted Voltage: 0.0
Qual Voltage: ZERO

Normal Current: 2.050776
Faulted Current: 0.0
Qual Current: ZERO

Readings at SENSOR-2C
Normal Voltage: 35.31241
Faulted Voltage: 0.585936
Qual Voltage: NORMAL

Normal Current: 2.148432

No: 6-6
8

Faulted Current: 0.585836
Qual Current: LOW

Readings at SENSOR-2D
Normal Voltage: 34.062412
Faulted Voltage: -0.585836
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: -0.585836
Qual Current: NORMAL

Readings at SENSOR-2B
Normal Voltage: 36.562405
Faulted Voltage: 0.292968
Qual Voltage: NORMAL

Normal Current: 1.074216
Faulted Current: 0.292968
Qual Current: LOW

Readings at SENSOR-2A
Normal Voltage: 36.24991
Faulted Voltage: 0.390624
Qual Voltage: NORMAL

Normal Current: 1.074216
Faulted Current: 0.390624
Qual Current: LOW

Readings at SENSOR-1F
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-1E
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-1C
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

```
Readings at SENSOR-1D
  Normal  Voltage:  5.6249857
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-1B
  Normal  Voltage:  0.9374976
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-1A
  Normal  Voltage:  0.9374976
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Program halted.
Ending run at 2/08/86 13:39:49.
 => Resetting ART...
Knowledge base has been reset.
 => Beginning run at 2/08/86 13:43:21: type ^C to halt.
Program halted.
Ending run at 2/08/86 14:00:20.
 =>  => Beginning run at 2/08/86 14:00:34: type ^C to halt.
 Initializing Circuit Description


 Create Errorbands


 Assigning Qualitative Values to Sensors


 Hypothesizing Faults:


Fault-Hypothesis:  Open-Circuit BUS-2B


Fault-Hypothesis:  Open-Circuit RELAY-2E


Fault-Hypothesis: Resistive-Short-Circuit SOLAR-ARRAY-2B


Fault-Hypothesis: Resistive-Short-Circuit SOLAR-ARRAY-2A


Fault-hyp: Open-Circuit RELAY-2E
Source Propagation from RELAY-2E to BUS-2B

Fault-hyp: OPEN-CIRCUIT RELAY-2E
      Propagation from BUS-2B to SENSOR-2C

    -hyp: OPEN-CIRCUIT RELAY-2E
```

F-6

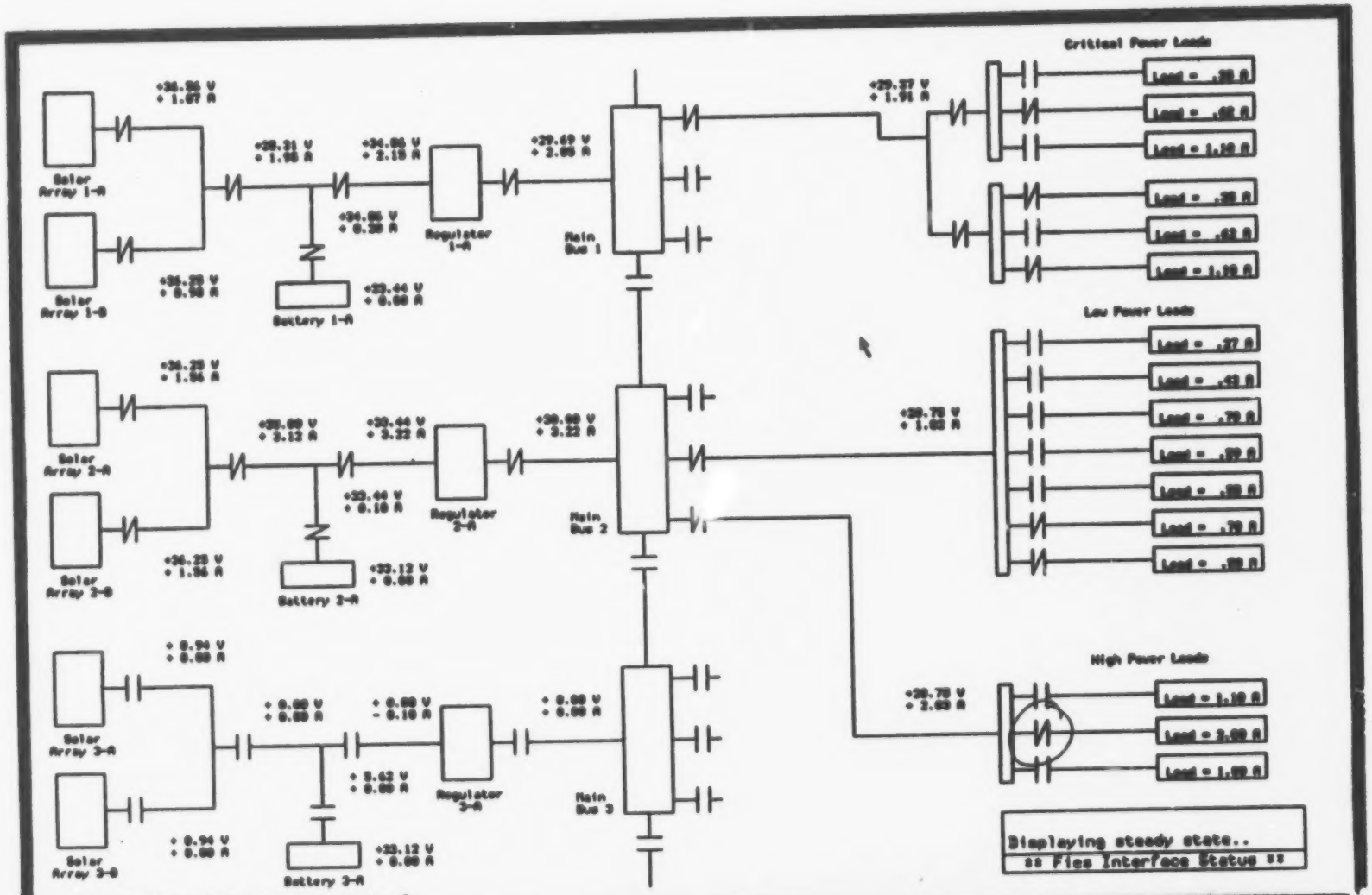
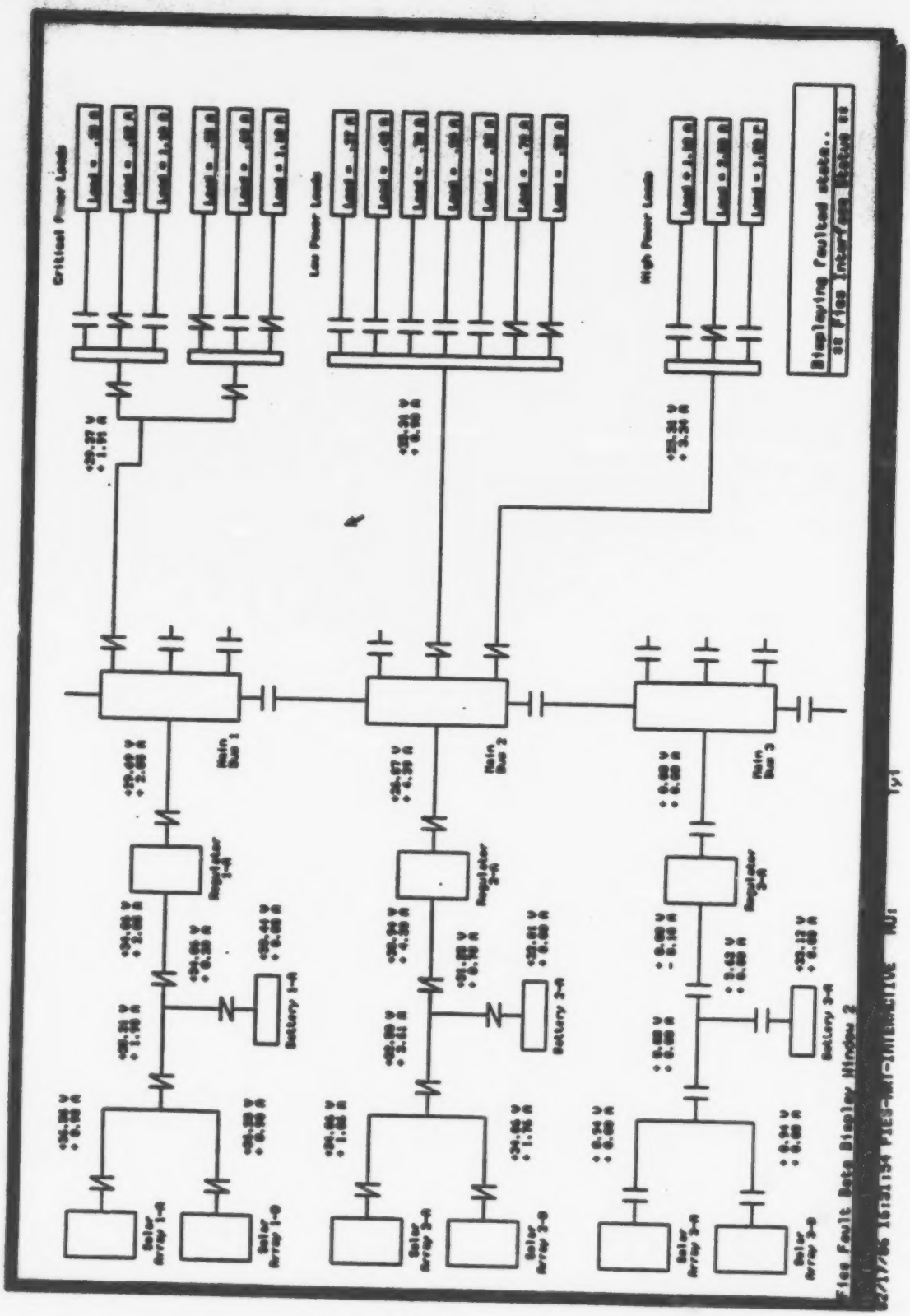
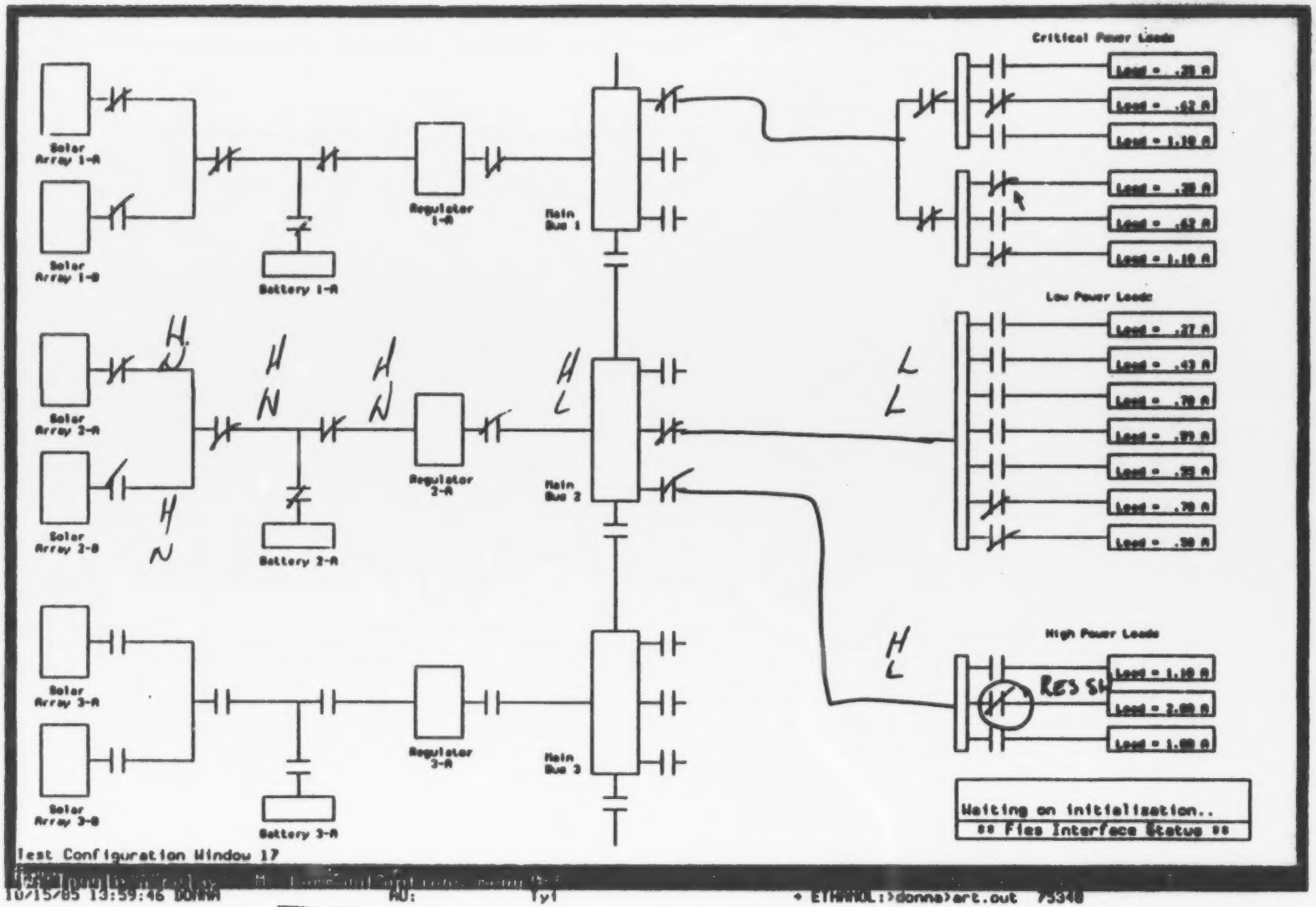


Fig. 27

ORIGINAL PAGE IS
OF POOR QUALITY


Critical Power Loads
Low Power Loads
High Power Loads
Displaying fault data..
Main Bus 1
Main Bus 2
Main Bus 3
Regulator 1-0
Regulator 2-0
Regulator 3-0
Battery 1-0
Battery 2-0
Battery 3-0
Solar Array 1-0
Solar Array 2-0
Solar Array 3-0
Fies Fault Data Display Window 2

Ex: F-9



RES SH ~~SATA~~ P ~~SATB~~ P ~~BATTA~~ P BUS2C P LOAD3M BUS3D

S-C LOAD 3M BUS 3D

RES 2M. DAT

F-7

3

ORIGINAL PAGE IS OF POOR QUALITY

```
=> pop
=> run
Beginning run at 2/17/86 16:32:49: type ^C to halt.
 Initializing Circuit Description

 Create Errorbands

 Assigning Qualitative Values to Sensors

 Hypothesizing Faults:

Fault-Hypothesis: Resistive-Short-Circuit BUS-2C

Fault-Hypothesis: Closed-Relay RELAY-3N

Fault-Hypothesis: Closed-Relay RELAY-3K

Fault-Hypothesis: Resistive-Short-Circuit BUS-3D

Fault-Hypothesis: Resistive-Short-Circuit LOAD-3M

Fault-Hypothesis: Short-Circuit BUS-3D

Fault-Hypothesis: Short-Circuit LOAD-3M

Reject Fault Hypothesis Short-Circuit at BUS-3D
Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-3G

Reject Fault Hypothesis Short-Circuit at LOAD-3M
Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-3G

Fault-hyp: Resistive-Short-Circuit BUS-2C
Source Propagation from BUS-2C to SENSOR-2F

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2F
Source Propagation from SENSOR-2F to RELAY-2F

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from RELAY-2F to REGULATOR-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from REGULATOR-2A to SENSOR-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2E
Source Propagation from SENSOR-2E to RELAY-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from RELAY-2E to BUS-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from BUS-2B to SENSOR-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2D
Source Propagation from SENSOR-2D to RELAY-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from RELAY-2D to BATTERY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from BATTERY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
```

Fig: F-7

4

Source Propagation from BUS-2B to SENSOR-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2C
Source Propagation from SENSOR-2C to RELAY-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: Resistive-Short-Circuit BUS-2C
Sink Propagation from BUS-2C to RELAY-2H

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Sink Propagation from RELAY-2H to SENSOR-2G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Predicted Current Effect: LOW Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Sink Propagation from BUS-2D to RELAY-2Q

FFFault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
End-Sink Propagation from LOAD-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Sink Propagation from BUS-2D to RELAY-2R

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Sink Propagation from RELAY-2R to LOAD-2R

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
End-Sink Propagation from LOAD-2R

Fault-hyp: Resistive-Short-Circuit BUS-2C
Sink Propagation from BUS-2C to RELAY-2J

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-2C
Sink Propagation from RELAY-2J to SENSOR-3G

Reject Fault Hypothesis RESISTIVE-SHORT-CIRCUIT at BUS-2C
Reason: Predicted Effect Is Inconsistent at Sensor SENSOR-3G
Predicted Value: LOW
[illegible]ser Value: HIGH

[illegible]p: Resistive-Short-Circuit BUS-3D

5

Source Propagation from BUS-3D to SENSOR-3G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-3G
Source Propagation from SENSOR-3G to RELAY-2J

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2J to BUS-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source-Propagation from BUS-2C to SENSOR-2F
Sink-Propagation from BUS-2C to RELAY-2H

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2F
Source Propagation from SENSOR-2F to RELAY-2F

FFFault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2F to REGULATOR-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from REGULATOR-2A to SENSOR-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2E
Source Propagation from SENSOR-2E to RELAY-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2E to BUS-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2B to SENSOR-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2D
Source Propagation from SENSOR-2D to RELAY-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2D to BATTERY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Source Propagation from BATTERY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2B to SENSOR-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2C
Source Propagation from SENSOR-2C to RELAY-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
SSSource Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

F-7

6

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2H to SENSOR-2G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: LOW Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from BUS-2D to RELAY-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from BUS-2D to RELAY-2R

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2R to LOAD-2R

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-2R

FFFault-hyp: Resistive-Short-Circuit BUS-3D
Sink Propagation from BUS-3D to RELAY-3M

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-3M to LOAD-3M

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-3M

Fault-hyp: Resistive-Short-Circuit LOAD-3M
Source Propagation from LOAD-3M to RELAY-3M

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from RELAY-3M to BUS-3D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from BUS-3D to SENSOR-3G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-3G
Source Propagation from SENSOR-3G to RELAY-2J

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from RELAY-2J to BUS-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source-Propagation from BUS-2C to SENSOR-2F
Sink-Propagation from BUS-2C to RELAY-2H

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2F
Source Propagation from SENSOR-2F to RELAY-2F

F-7

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from RELAY-2F to REGULATOR-2A

7

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from REGULATOR-2A to SENSOR-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2E
Source Propagation from SENSOR-2E to RELAY-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
[illegible]e Propagation from RELAY-2E to BUS-2B

[illegible]yp: RESISTIVE-SHORT-CIRCUIT LOAD-3M

Source Propagation from BUS-2B to SENSOR-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2D
Source Propagation from SENSOR-2D to RELAY-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from RELAY-2D to BATTERY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
End-Source Propagation from BATTERY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from BUS-2B to SENSOR-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2C
Source Propagation from SENSOR-2C to RELAY-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Sink Propagation from RELAY-2N to SENSOR-2G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Predicted Current Effect: LOW Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Sink Propagation from BUS-2D to RELAY-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
End-Sink Propagation from LOAD-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Sink Propagation from BUS-2D to RELAY-2R

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
Sink Propagation from RELAY-2R to LOAD-2R

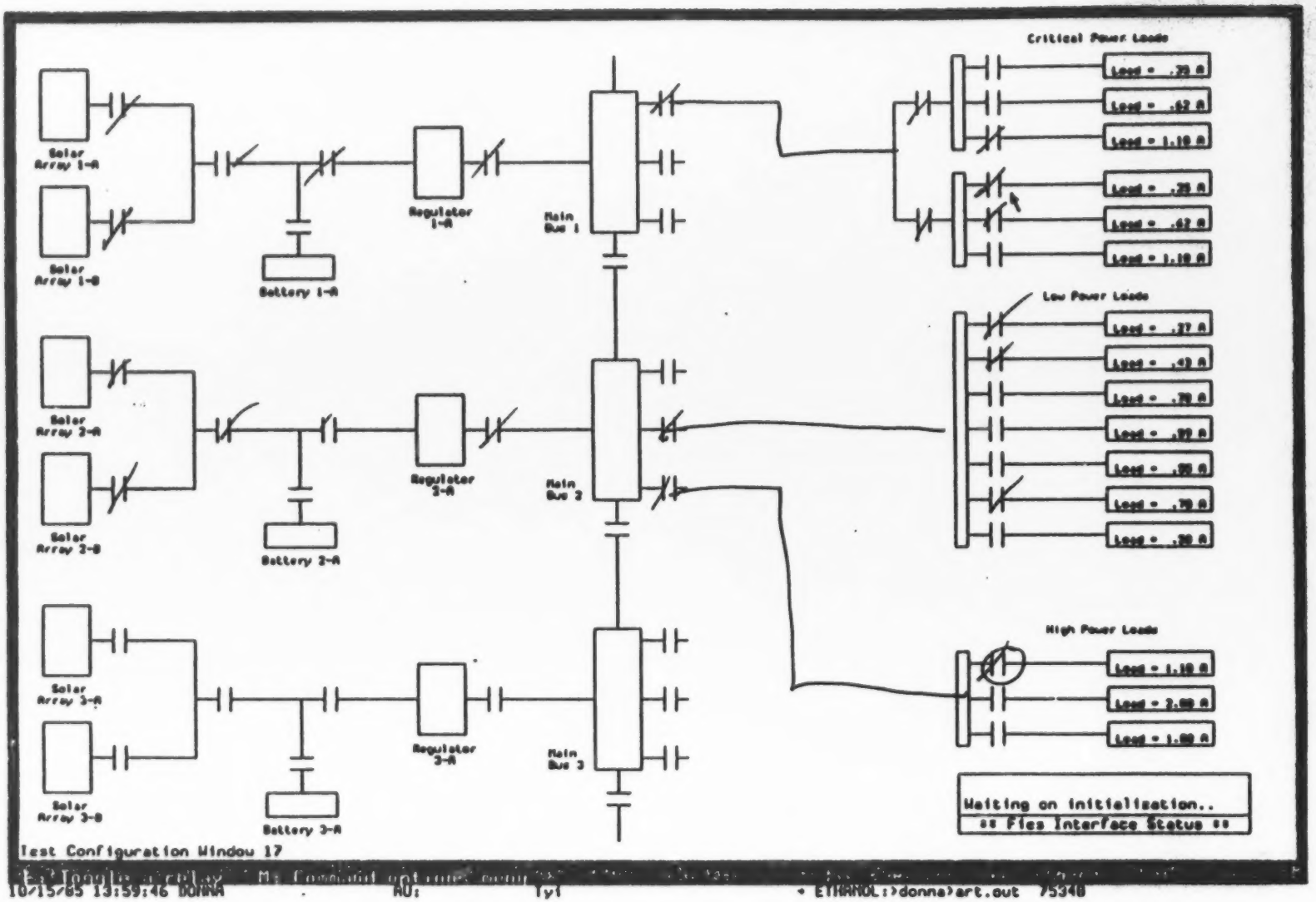
Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3M
End-Sink Propagation from LOAD-2R

:ing All Initial Fault Hypotheses:

F-7

D

```
      RESISTIVE-SHORT-CIRCUIT BUS-2C

      CLOSED-RELAY RELAY-3N

      CLOSED-RELAY RELAY-3K

      RESISTIVE-SHORT-CIRCUIT BUS-3D

        RESISTIVE-SHORT-CIRCUIT LOAD-3M

      SHORT-CIRCUIT BUS-3D

      SHORT-CIRCUIT LOAD-3M


Reporting Final Fault Hypotheses:

      CLOSED-RELAY RELAY-3N

      CLOSED-RELAY RELAY-3K

      RESISTIVE-SHORT-CIRCUIT BUS-3D

      RESISTIVE-SHORT-CIRCUIT LOAD-3M

Program halted.
Ending run at 2/17/86 16:33:51.
 => watch
 => dribble
```



short 9 dat

Fig. F-8

ORIGINAL PAGE IS
OF POOR QUALITY

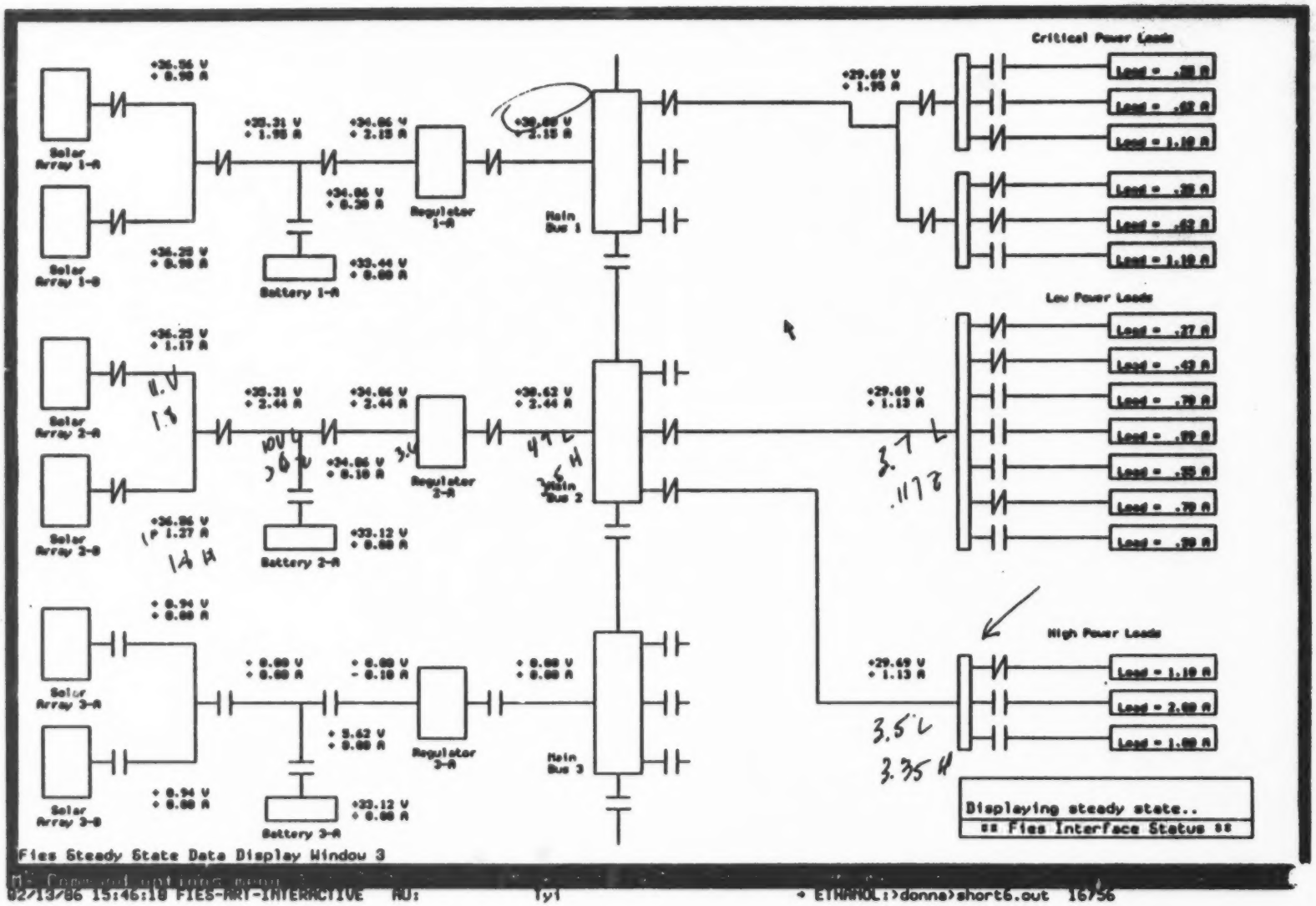
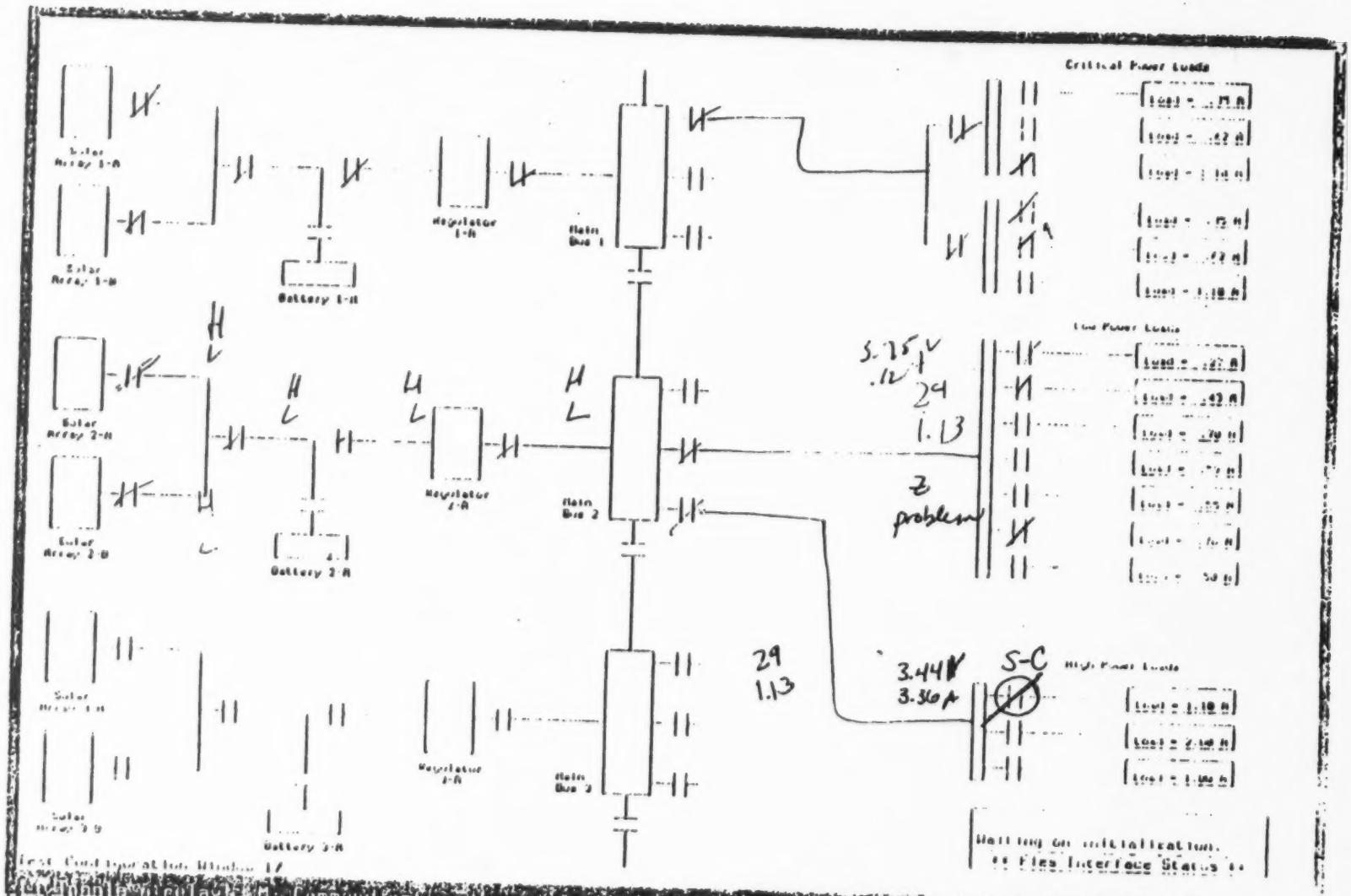

Critical Power Loads
Low Power Loads
High Power Loads
Solar Array 1-A
Solar Array 1-B
Solar Array 2-A
Solar Array 2-B
Solar Array 3-A
Solar Array 3-B
Battery 1-A
Battery 2-A
Battery 3-A
Regulator 1-A
Regulator 2-A
Regulator 3-A
Main Bus 1
Main Bus 2
Main Bus 3
Displaying steady state..
** Fies Interface Status **
Fies Steady State Data Display Window 3
02/13/86 15:46:18 FIES-ART-INTERACTIVE AU: Tyl
ETHANOL:>donna>short6.out 16/56

Fig. 8B



S-C Bus 3D Load 3K Res-SH LOAD 3K BUS 3D CLOSED Relay 3H 3H

Direct Short 9.

Fig. F-8

3

ORIGINAL PAGE IS OF POOR QUALITY

```
Readings at SENSOR-1D
  Normal  Voltage:  34.062412
  Faulted Voltage:  34.062412
  Qual Voltage:     NORMAL

  Normal Current:   0.195312
  Faulted Current:  0.097656
  Qual Current:     NORMAL


Readings at SENSOR-1B
  Normal  Voltage:  36.24991
  Faulted Voltage:  36.24991
  Qual Voltage:     NORMAL

  Normal Current:   0.878904
  Faulted Current:  0.97656
  Qual Current:     NORMAL


Readings at SENSOR-1A
  Normal  Voltage:  36.562405
  Faulted Voltage:  36.562405
  Qual Voltage:     NORMAL

  Normal Current:   1.074216
  Faulted Current:  1.074216
  Qual Current:     NORMAL


Program halted.
Ending run at 2/13/86 15:44:27.
 => reset
Resetting ART...
Knowledge base has been reset.
 => run
Beginning run at 2/13/86 15:45:17: type ^C to halt.
Program halted.
Ending run at 2/13/86 15:47:03.
 => Beginning run at 2/13/86 15:47:05: type ^C to halt.
 Initializing Circuit Description


 Create Errorbands


 Assigning Qualitative Values to Sensors


 Hypothesizing Faults:


Fault-Hypothesis: Closed-Relay RELAY-3N


Fault-Hypothesis: Closed-Relay RELAY-3M


Fault-Hypothesis: Resistive-Short-Circuit BUS-3D


Fault-Hypothesis: Resistive-Short-Circuit LOAD-3K


Fault-Hypothesis: Short-Circuit BUS-3D


Fault-Hypothesis: Short-Circuit LOAD-3K


        Fault Hypothesis Resistive-Short-Circuit at BUS-3D
```

F-8

4

Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-3G

Reject Fault Hypothesis Closed-Relay at RELAY-3M
Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-3G

Reject Fault Hypothesis Closed-Relay at RELAY-3N
Reason: Predicted Voltage Effect Is Inconsistent at Sensor SENSOR-3G

Fault-hyp: Short-Circuit BUS-3D
Source Propagation from BUS-3D to SENSOR-3G

Fault-hyp: SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-3G
Source Propagation from SENSOR-3G to RELAY-2J

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2J to BUS-2C

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source-Propagation from BUS-2C to SENSOR-2F
Sink-Propagation from BUS-2C to RELAY-2H

Fault-hyp: SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2F
Source Propagation from SENSOR-2F to RELAY-2F

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2F to REGULATOR-2A

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from REGULATOR-2A to SENSOR-2E

Fault-hyp: SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2E
Source Propagation from SENSOR-2E to RELAY-2E

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2E to BUS-2B

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2B to SENSOR-2C

Fault-hyp: SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2C
Source Propagation from SENSOR-2C to RELAY-2C

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: SHORT-CIRCUIT BUS-3D
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

F-8
5

Fault-hyp: SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: SHORT-CIRCUIT BUS-3D
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: SHORT-CIRCUIT BUS-3D

Sink Propagation from RELAY-2H to SENSOR-2G

Fault-hyp: SHORT-CIRCUIT BUS-3D
Predicted Current Effect: LOW Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: SHORT-CIRCUIT BUS-3D
Sink Propagation from BUS-2D to RELAY-2K

Fault-hyp: SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2K to LOAD-2K

Fault-hyp: SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-2K

Fault-hyp: SHORT-CIRCUIT BUS-3D
Sink Propagation from BUS-2D to RELAY-2M

Fault-hyp: SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2M to LOAD-2M

Fault-hyp: SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-2M

Fault-hyp: SHORT-CIRCUIT BUS-3D
Sink Propagation from BUS-2D to RELAY-2Q

Fault-hyp: SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-2Q

Fault-hyp: Short-Circuit BUS-3D
Sink Propagation from BUS-3D to RELAY-3K

Fault-hyp: SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-3K to LOAD-3K

Fault-hyp: SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-3K

Fault-hyp: Short-Circuit LOAD-3K
Source Propagation from LOAD-3K to RELAY-3K

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-3K to BUS-3D

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from BUS-3D to SENSOR-3G

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-3G
Source Propagation from SENSOR-3G to RELAY-2J

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2J to BUS-2C

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source-Propagation from BUS-2C to SENSOR-2F
Sink-Propagation from BUS-2C to RELAY-2H

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2F
Source Propagation from SENSOR-2F to RELAY-2F

F-8
6

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2F to REGULATOR-2A

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from REGULATOR-2A to SENSOR-2E

[illegible]hyp: SHORT-CIRCUIT LOAD-3K
[illegible]ed Current Effect: HIGH Consistent at Sensor SENSOR-2E
[illegible] Propagation from SENSOR-2E to RELAY-2E

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2E to BUS-2B

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from BUS-2B to SENSOR-2C

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2C
Source Propagation from SENSOR-2C to RELAY-2C

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: SHORT-CIRCUIT LOAD-3K
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: SHORT-CIRCUIT LOAD-3K
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Sink Propagation from RELAY-2H to SENSOR-2G

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: LOW Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Sink Propagation from BUS-2D to RELAY-2K

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Sink Propagation from RELAY-2K to LOAD-2K

Fault-hyp: SHORT-CIRCUIT LOAD-3K
End-Sink Propagation from LOAD-2K

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Sink Propagation from BUS-2D to RELAY-2M

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Sink Propagation from RELAY-2M to LOAD-2M

Fault-hyp: SHORT-CIRCUIT LOAD-3K
End-Sink Propagation from LOAD-2M

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Sink Propagation from BUS-2D to RELAY-2Q

Fault-hyp: SHORT-CIRCUIT LOAD-3K
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: SHORT-CIRCUIT LOAD-3K
End-Sink Propagation from LOAD-2Q

Fault-hyp: Resistive-Short-Circuit BUS-3D

F-8

7

Source Propagation from BUS-3D to SENSOR-3G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-3G
Source Propagation from SENSOR-3G to RELAY-2J

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2J to BUS-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source-Propagation from BUS-2C to SENSOR-2F
Sink-Propagation from BUS-2C to RELAY-2H

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2F
Source Propagation from SENSOR-2F to RELAY-2F

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2F to REGULATOR-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from REGULATOR-2A to SENSOR-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2E
Source Propagation from SENSOR-2E to RELAY-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2E to BUS-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2B to SENSOR-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2C
Source Propagation from SENSOR-2C to RELAY-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2H to SENSOR-2G

Fig 8

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Predicted Current Effect: LOW Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from BUS-2D to RELAY-2K

```
Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2K to LOAD-2K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-2K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from BUS-2D to RELAY-2M

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2M to LOAD-2M

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-2M

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from BUS-2D to RELAY-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-2Q

Fault-hyp: Resistive-Short-Circuit BUS-3D
Sink Propagation from BUS-3D to RELAY-3K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
Sink Propagation from RELAY-3K to LOAD-3K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT BUS-3D
End-Sink Propagation from LOAD-3K

Fault-hyp: Resistive-Short-Circuit LOAD-3K
Source Propagation from LOAD-3K to RELAY-3K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-3K to BUS-3D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from BUS-3D to SENSOR-3G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-3G
Source Propagation from SENSOR-3G to RELAY-2J

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2J to BUS-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source-Propagation from BUS-2C to SENSOR-2F
Sink-Propagation from BUS-2C to RELAY-2H

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2F
Source Propagation from SENSOR-2F to RELAY-2F

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2F to REGULATOR-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from REGULATOR-2A to SENSOR-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2E
Source Propagation from SENSOR-2E to RELAY-2E

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2E to BUS-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from BUS-2B to SENSOR-2C

    -hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
    cted Current Effect: HIGH Consistent at Sensor SENSOR-2C
```

F-8

9

Source Propagation from SENSOR-2C to RELAY-2C

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: HIGH Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Sink Propagation from RELAY-2H to SENSOR-2G

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Predicted Current Effect: LOW Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Sink Propagation from BUS-2D to RELAY-2K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Sink Propagation from RELAY-2K to LOAD-2K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
End-Sink Propagation from LOAD-2K

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Sink Propagation from BUS-2D to RELAY-2M

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Sink Propagation from RELAY-2M to LOAD-2M

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
End-Sink Propagation from LOAD-2M

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Sink Propagation from BUS-2D to RELAY-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
Sink Propagation from RELAY-2Q to LOAD-2Q

Fault-hyp: RESISTIVE-SHORT-CIRCUIT LOAD-3K
End-Sink Propagation from LOAD-2Q

Reporting All Initial Fault Hypotheses:

f-8

SHORT-CIRCUIT LOAD-3K 10

SHORT-CIRCUIT BUS-3D

RESISTIVE-SHORT-CIRCUIT LOAD-3K

RESISTIVE-SHORT-CIRCUIT BUS-3D

CLOSED-RELAY RELAY-3M

CLOSED-RELAY RELAY-3N

Reporting Final Fault Hypotheses:

SHORT-CIRCUIT LOAD-3K

SHORT-CIRCUIT BUS-3D

RESISTIVE-SHORT-CIRCUIT LOAD-3K

RESISTIVE-SHORT-CIRCUIT BUS-3D

Readings at SENSOR-3Z
Normal Voltage: 33.124916
Faulted Voltage: 33.124916
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2Z
Normal Voltage: 33.124916
Faulted Voltage: 33.124916
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-1Z
Normal Voltage: 33.437412
Faulted Voltage: 33.437412
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3G
Normal Voltage: 29.687424
Faulted Voltage: 3.4374912
Qual Voltage: LOW

Normal Current: 1.1320096
Faulted Current: 3.3593664
Qual Current: HIGH

Readings at SENSOR-2G
Normal Voltage: 29.687424
Faulted Voltage: 3.7499605
Qual Voltage: LOW

Normal Current: 1.1320096
Faulted Current: 0.1171872
Qual Current: ZERO

F-8

11

Readings at SENSOR-1G
Normal Voltage: 29.687424
Faulted Voltage: 29.374824
Qual Voltage: NORMAL

Normal Current: 1.95312
Faulted Current: 1.9140576
Qual Current: NORMAL

Readings at SENSOR-3F
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3E
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: -0.097656
Faulted Current: -0.097656
Qual Current: NORMAL

Readings at SENSOR-3C
Normal Voltage: 0.0
Faulted Voltage: 0.0
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-3D
Normal Voltage: 5.6249857
Faulted Voltage: 5.6249857
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.097656
Qual Current: NORMAL

Readings at SENSOR-3B
Normal Voltage: 0.9374976
Faulted Voltage: 0.9374976
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Fig: F-1
12

gs at SENSOR-3A
al Voltage: 0.9374976

Faulted Voltage: 0.9374976
Qual Voltage: NORMAL

Normal Current: 0.0
Faulted Current: 0.0
Qual Current: NORMAL

Readings at SENSOR-2F
Normal Voltage: 30.624922
Faulted Voltage: 4.999987
Qual Voltage: LOW

Normal Current: 2.4414
Faulted Current: 3.515616
Qual Current: HIGH

Readings at SENSOR-2E
Normal Voltage: 34.062412
Faulted Voltage: 8.437478
Qual Voltage: LOW

Normal Current: 2.4414
Faulted Current: 3.613272
Qual Current: HIGH

Readings at SENSOR-2C
Normal Voltage: 35.31241
Faulted Voltage: 10.312473
Qual Voltage: LOW

Normal Current: 2.4414
Faulted Current: 3.613272
Qual Current: HIGH

Readings at SENSOR-2D
Normal Voltage: 34.062412
Faulted Voltage: 8.749977
Qual Voltage: LOW

Normal Current: 0.097656
Faulted Current: 0.0
Qual Current: ZERO

Readings at SENSOR-2B
Normal Voltage: 36.562405
Faulted Voltage: 11.56247
Qual Voltage: LOW

Normal Current: 1.2695279
Faulted Current: 1.855464
Qual Current: HIGH

F-8

13

Readings at SENSOR-2A
Normal Voltage: 36.24981
Faulted Voltage: 11.56247
Qual Voltage: LOW

Current: 1.171872

```
  Faulted Current:  1.757808
  Qual Current:     HIGH


Readings at SENSOR-1F
  Normal  Voltage:  29.999924
  Faulted Voltage:  29.999924
  Qual Voltage:     NORMAL

  Normal Current:   2.148432
  Faulted Current:  2.148432
  Qual Current:     NORMAL


Readings at SENSOR-1E
  Normal  Voltage:  34.062412
  Faulted Voltage:  34.062412
  Qual Voltage:     NORMAL

  Normal Current:   2.148432
  Faulted Current:  2.148432
  Qual Current:     NORMAL


Readings at SENSOR-1C
  Normal  Voltage:  35.31241
  Faulted Voltage:  35.31241
  Qual Voltage:     NORMAL

  Normal Current:   1.95312
  Faulted Current:  1.95312
  Qual Current:     NORMAL


Readings at SENSOR-1D
  Normal  Voltage:  34.062412
  Faulted Voltage:  34.062412
  Qual Voltage:     NORMAL

  Normal Current:   0.195312
  Faulted Current:  0.195312
  Qual Current:     NORMAL


Readings at SENSOR-1B
  Normal  Voltage:  38.24991
  Faulted Voltage:  38.24991
  Qual Voltage:     NORMAL

  Normal Current:   0.97656
  Faulted Current:  0.97656
  Qual Current:     NORMAL


Readings at SENSOR-1A
  Normal  Voltage:  38.562405
  Faulted Voltage:  38.562405
  Qual Voltage:     NORMAL

  Normal Current:   0.97656
  Faulted Current:  0.97656
  Qual Current:     NORMAL
```

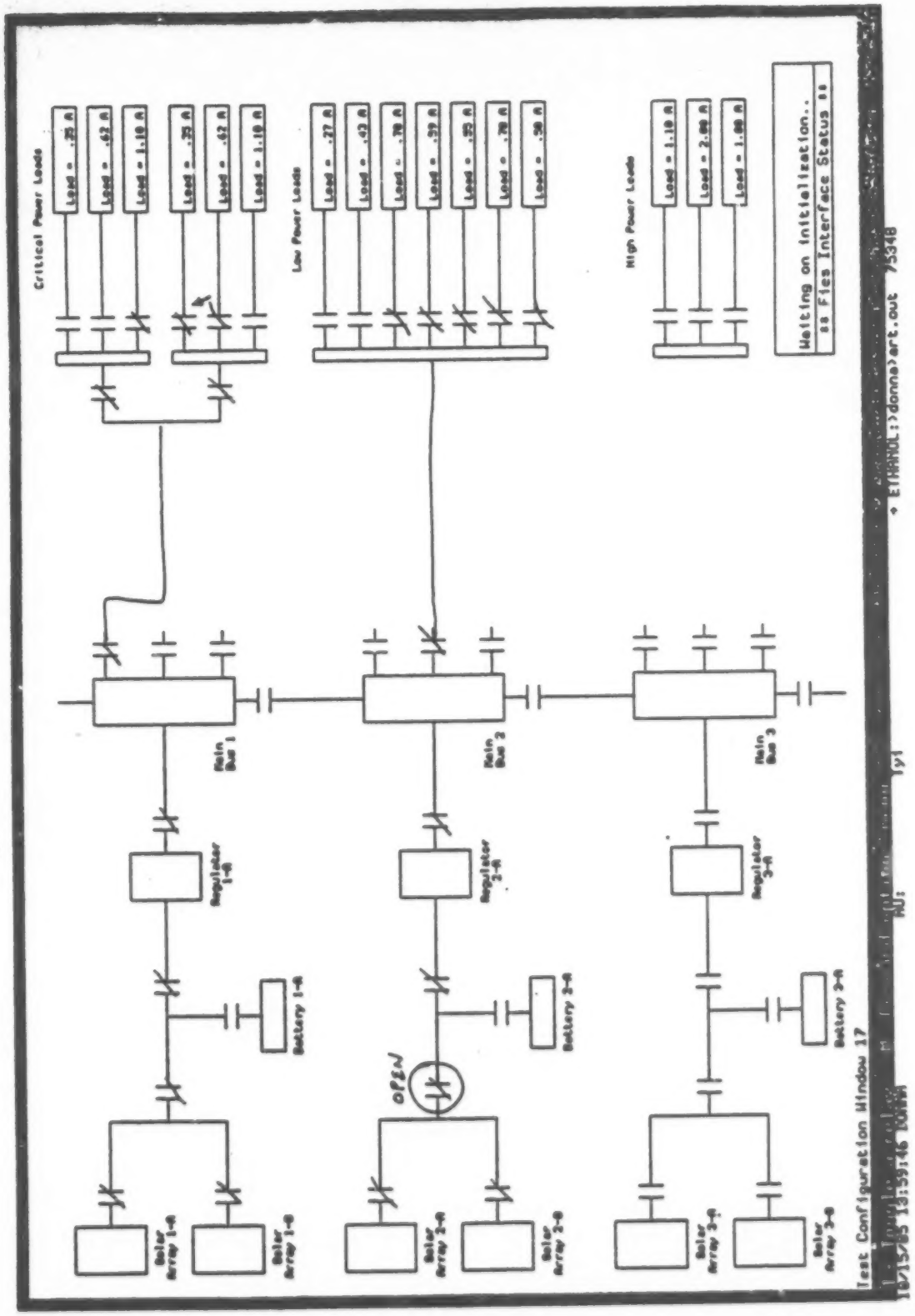
F-8

14

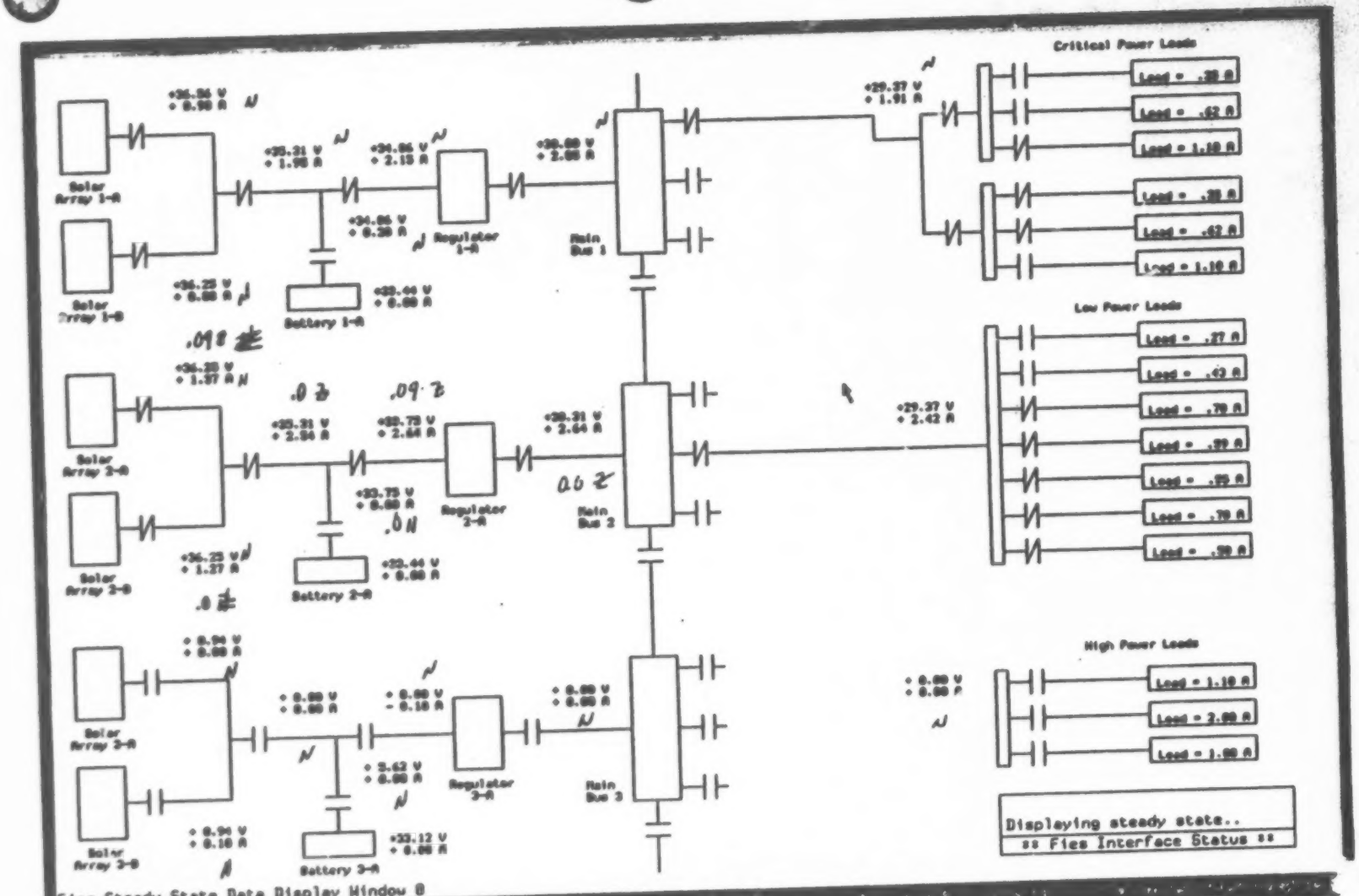
```
Program halted.
Ending run at 2/13/86 15:49:12.
 => watch
 => dribble
```

Fig. F-8

ORIGINAL PAGE IS
OF POOR QUALITY



open5.dat



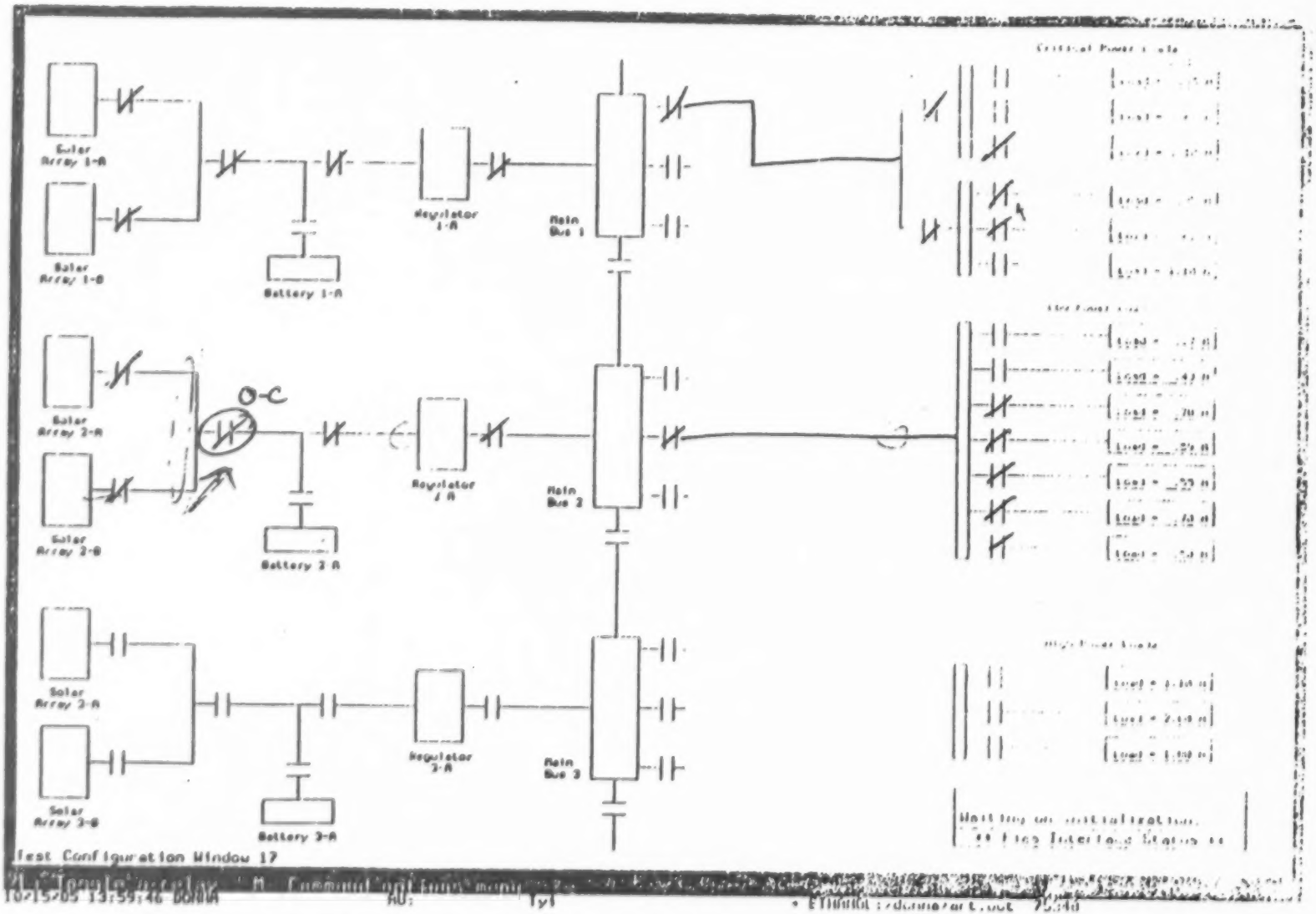
02/07/86 16:06:58 FIES-ARY-INTERACTIVE AU: 1y1

open5.dat

Fig 2

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY



ResSh ~~SA1A P~~ ~~1B P~~ 2A P 2B P

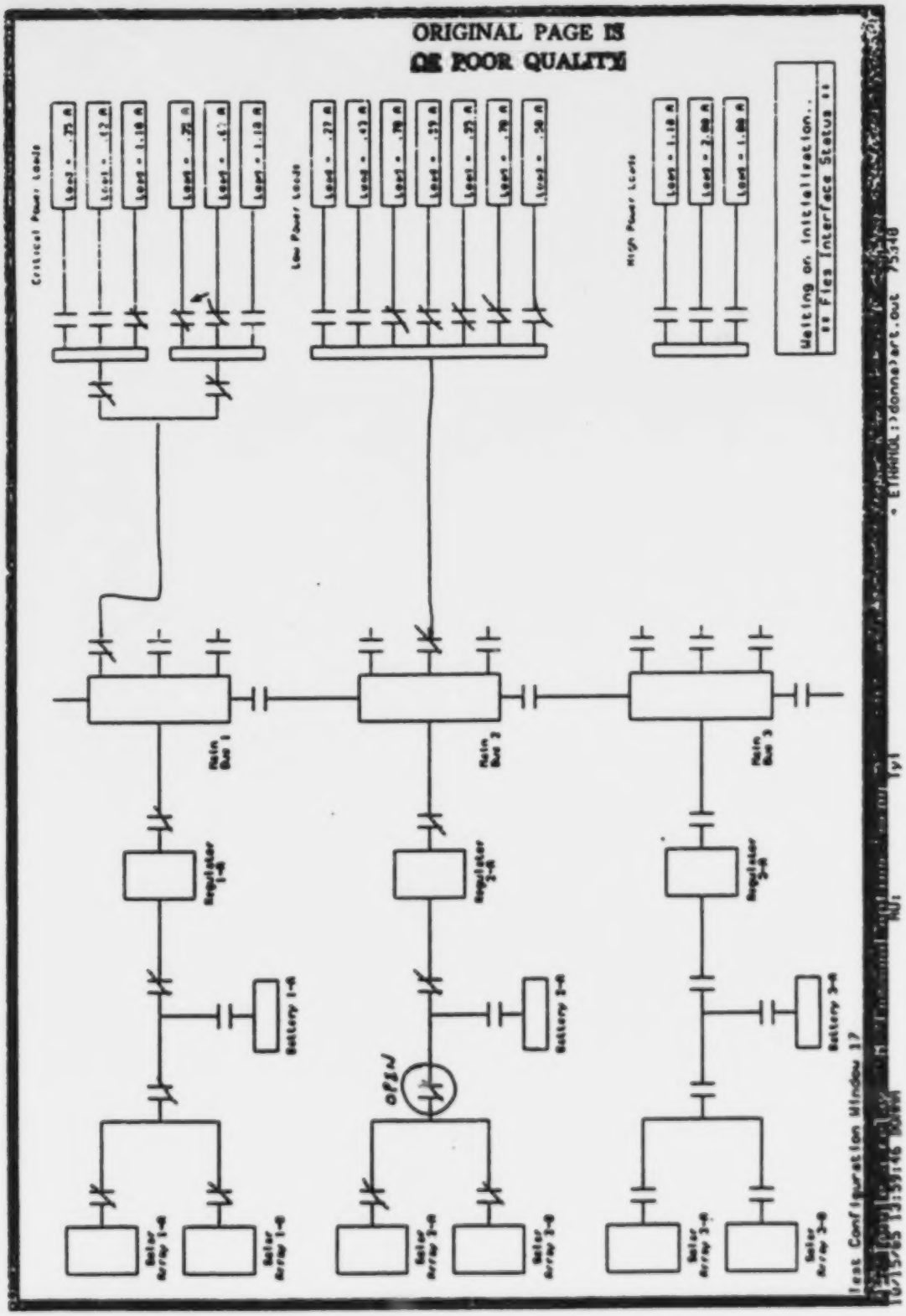
O-C SA2A P 2B P

Relay 2C Bus2A

OPEN 5.DAT

Fig. 3 F-9

ORIGINAL PAGE IS
OF POOR QUALITY

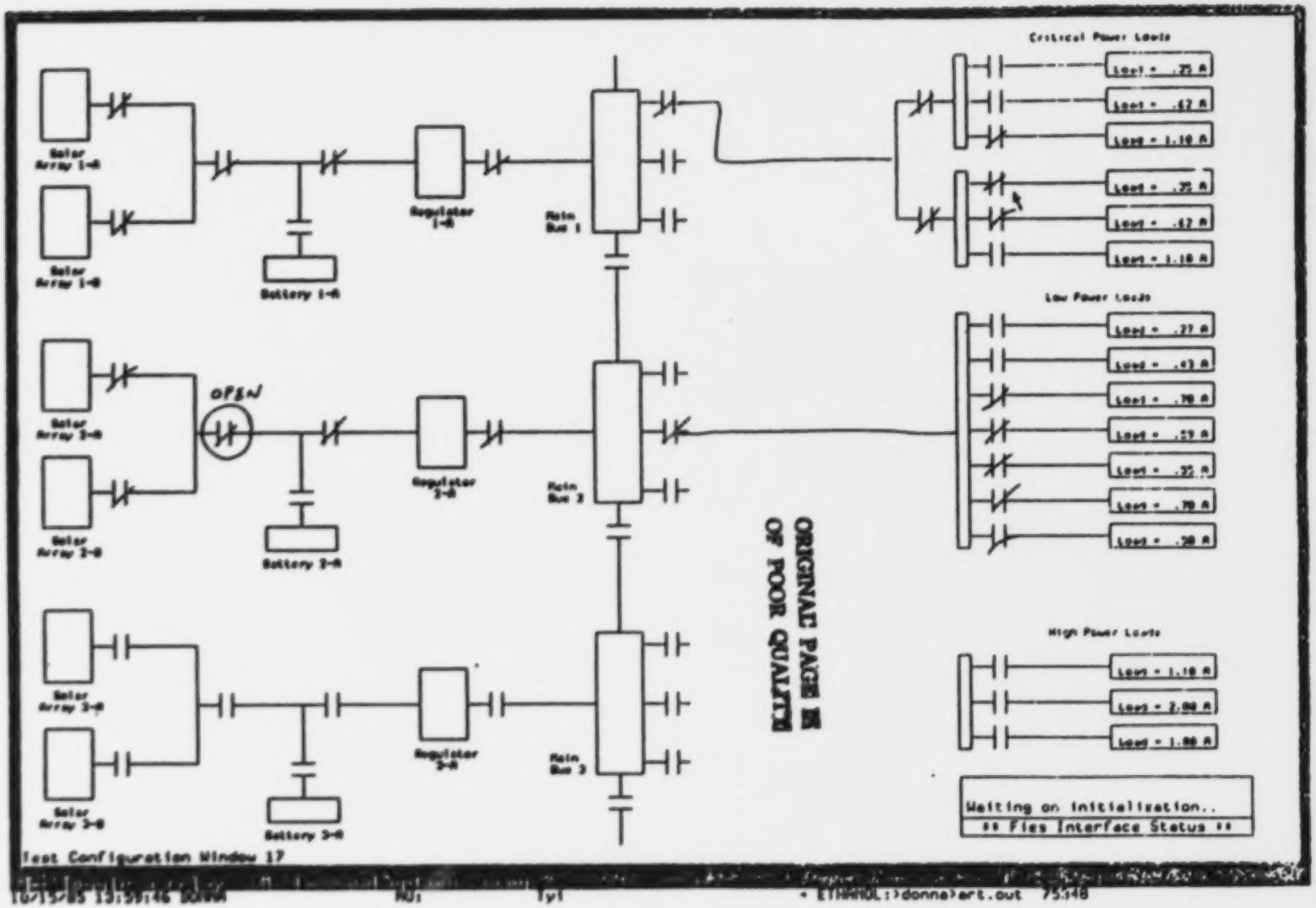


open5.dat

C-3

Ex: FA

4



open5.dat

ORIGINAL PAGE IS
OF POOR QUALITY

```
Fault-hyp: Open-Circuit RELAY-2C
Source Propagation from RELAY-2C to BUS-2A

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Source Propagation from BUS-2A to SENSOR-2B

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2B
Source Propagation from SENSOR-2B to RELAY-2B

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Source Propagation from RELAY-2B to SOLAR-ARRAY-2B

Fault-hyp: OPEN-CIRCUIT RELAY-2C
End-Source Propagation from SOLAR-ARRAY-2B

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Source Propagation from BUS-2A to SENSOR-2A

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2A
Source Propagation from SENSOR-2A to RELAY-2A

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Source Propagation from RELAY-2A to SOLAR-ARRAY-2A

Fault-hyp: OPEN-CIRCUIT RELAY-2C
End-Source Propagation from SOLAR-ARRAY-2A

Fault-hyp: Open-Circuit RELAY-2C
Sink Propagation from RELAY-2C to SENSOR-2C


Fault-hyp: OPEN-CIRCUIT RELAY-2C
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2C
Sink Propagation from SENSOR-2C to BUS-2B

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Sink Propagation from BUS-2B to RELAY-2E

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Sink Propagation from RELAY-2E to SENSOR-2E


Fault-hyp: OPEN-CIRCUIT RELAY-2C
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2E
Sink Propagation from SENSOR-2E to REGULATOR-2A

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Sink Propagation from REGULATOR-2A to RELAY-2F

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Sink Propagation from RELAY-2F to SENSOR-2F


Fault-hyp: OPEN-CIRCUIT RELAY-2C
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2F
Sink Propagation from SENSOR-2F to BUS-2C

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Sink Propagation from BUS-2C to RELAY-2H

Fault-hyp: OPEN-CIRCUIT RELAY-2C
Sink Propagation from RELAY-2H to SENSOR-2G


Fault-hyp: OPEN-CIRCUIT RELAY-2C
Predicted Current Effect: ZERO Consistent at Sensor SENSOR-2G
Sink Propagation from SENSOR-2G to BUS-2D

ult-hyp: OPEN-CIRCUIT RELAY-2C
k Propagation from BUS-2D to RELAY-2N

-hyp: OPEN-CIRCUIT RELAY-2C
. Propagation from RELAY-2N to LOAD-2N
```

F1

6

Propagation from BUS-2A to SENSOR-2A

Fault Hypothesis RESISTIVE-SHORT-CIRCUIT at SOLAR-ARRAY-2B
Predicted Effect Is Inconsistent at Sensor SENSOR-2A
:dicted Value: HIGH
:sor Value: ZERO

ng All Initial Fault Hypotheses:

'EN-CIRCUIT SOLAR-ARRAY-2A

:SISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2A

'EN-CIRCUIT SOLAR-ARRAY-2B

:SISTIVE-SHORT-CIRCUIT SOLAR-ARRAY-2B

'EN-CIRCUIT RELAY-2C

'EN-CIRCUIT BUS-2A

ng Final Fault Hypotheses:

'PEN-CIRCUIT RELAY-2C

PEN-CIRCUIT BUS-2A

SENSOR-3Z
oltage: 33.124816
ed Voltage: 0.0
Voltage: NORMAL

l Current: 0.0
ed Current: 0.0
Current: NORMAL

s at SENSOR-2Z
l Voltage: 33.437412
ed Voltage: 0.0
Voltage: NORMAL

l Current: 0.0
ed Current: 0.0
Current: NORMAL

s at SENSOR-1Z
l Voltage: 33.437412
ed Voltage: 0.0
Voltage: NORMAL

l Current: 0.0
ed Current: 0.0
Current: NORMAL

SENSOR-3S
l Voltage: 0.0
ed Voltage: -0.0300024
Voltage: NORMAL

l Current: 0.0
ed Current: -0.0300024
Current: NORMAL

F-9
6

```
  Normal  Voltage:  0.9374978
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   0.097056
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-3A
  Normal  Voltage:  0.9374978
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-2F
  Normal  Voltage:  30.312422
  Faulted Voltage:  0.0
  Qual Voltage:     ZERO

  Normal Current:   2.6367118
  Faulted Current:  0.0
  Qual Current:     ZERO


Readings at SENSOR-2E
  Normal  Voltage:  33.748012
  Faulted Voltage:  0.097056
  Qual Voltage:     ZERO

  Normal Current:   2.6367118
  Faulted Current:  0.097056
  Qual Current:     ZERO


Readings at SENSOR-2C
  Normal  Voltage:  36.31241
  Faulted Voltage:  0.0
  Qual Voltage:     ZERO

  Normal Current:   2.5360606
  Faulted Current:  0.0
  Qual Current:     ZERO


Readings at SENSOR-2D
  Normal  Voltage:  33.748012
  Faulted Voltage:  0.0
  Qual Voltage:     ZERO

  Normal Current:   0.0
  Faulted Current:  0.0
  Qual Current:     NORMAL


Readings at SENSOR-2B
  Normal  Voltage:  36.24001
  Faulted Voltage:  0.0
  Qual Voltage:     NORMAL
```

Normal Current: 1.2995279
Faulted Current: 0.0
Qual Current: ZERO

Readings at SENSOR-2A
Normal Voltage: 36.24001
Faulted Voltage: 0.007858
Qual Voltage: NORMAL

Normal Current: 1.3671839
Faulted Current: 0.007858
Qual Current: ZERO

Readings at SENSOR-1F
Normal Voltage: 29.999024
Faulted Voltage: 2.050778
Qual Voltage: NORMAL

Normal Current: 2.050778
Faulted Current: 2.050778
Qual Current: NORMAL

Readings at SENSOR-1E
Normal Voltage: 34.062412
Faulted Voltage: 2.050778
Qual Voltage: NORMAL

Normal Current: 2.148432
Faulted Current: 2.050778
Qual Current: NORMAL

Readings at SENSOR-1C
Normal Voltage: 35.31241
Faulted Voltage: 1.05312
Qual Voltage: NORMAL

Normal Current: 1.05312
Faulted Current: 1.05312
Qual Current: NORMAL

Readings at SENSOR-1D
Normal Voltage: 34.062412
Faulted Voltage: 0.195312
Qual Voltage: NORMAL

Normal Current: 0.195312
Faulted Current: 0.195312
Qual Current: NORMAL

Readings at SENSOR-1B
Normal Voltage: 36.24001
Faulted Voltage: 0.078804
Qual Voltage: NORMAL

Normal Current: 0.078804
Faulted Current: 0.078804
Qual Current: NORMAL

Ex: F-4
12

APPENDIX C

FIES II Software Design Specification

APPENDIX - C

FIES II Design Specification

Table of Contents

I. System Overview

- A. Initialization
- B. Fault Isolation

II. Microprocessor System Software Modular Profile

- A. Hierarchical decomposition
- B. Software Module descriptions

III. Symbolics System Software Modular Profile

- A. Hierarchical decomposition
- B. Software Module descriptions

IV. Symbolics Application expert system Software Modular Profile

- A. Hierarchical decomposition
- B. Software Module descriptions

V. Interface Description

- A. Overview
- B. Transmission senders
- C. Transmission receivers
- D. Data transmission formats
- E. Command functions summary

I. System Overview

This section is a high level description of the normal operation of the FIES system in terms of data flow across the interfaces of the system. For the purposes of this description four interfaces have been identified. These interfaces are breadboard to microprocessor, microprocessor to symbolics, symbolics to microprocessor and microprocessor to breadboard.

There are two distinct processing phases in the FIES system. The first phase is initialization phase which begins with breadboard, microprocessor and symbolics power up, includes user selection and download of an initial configuration for the breadboard and concludes when the microprocessor notifies the Symbolics that the breadboard has achieved steady state configuration. The second phase begins with fault insertion into a steady state configuration and concludes with the successful isolation and correction of a fault by the expert system.

A. Initialization

1.0 Power up microprocessor and breadboard

Power is applied simultaneously to the microprocessor and the breadboard by throwing the "power on" circuit breaker on the main control panel. This results in self test by both the microprocessor and B/B. Following successful self test the breadboard is placed in a 'safe' default configuration.

2.0 Microprocessor Initial Program Load

Following successful self test, control is transferred to a command file that loads the microprocessor FIES application software. Interrupts from the breadboard are enabled and the microprocessor begins handling interrupts from the breadboard and storing data in memory tables. This data acquisition cycle repeats itself every 3 milliseconds. Interrupts from the Symbolics are also enabled.

3.0 Power Up Symbolics

Power on to the Symbolics results in self test by the Symbolics.

4.0 Symbolics system Software Initial Program Load

Following successful self test, Symbolics system software is loaded into memory. Communication between the Symbolics and the microcomputer is now established and is confirmed by a control sequence handshake.

5.0 Initial Configuration Load

Following successful Symbolics IPL, the user is prompted with a configuration load option menu. Following selection of a configuration option, the Symbolics downloads the selected configuration to the microprocessor. Interrupts from the microprocessor are enabled and the Symbolics waits for the microprocessor upload of the steady state data values.

6.0 Microprocessor Configuration Load

Configuration load from the Symbolics interrupts the microprocessor. The interrupt is handled by an Interrupt Service Request and control is passed to a program that sets the breadboard to the initial configuration and records the initial configuration and the error band data in memory tables.

7.0 Microprocessor Steady State Detect

The microprocessor then begins sampling the breadboard a 3 ms. intervals until steady state is detected. When steady state is detected the steady state values are uploaded to the Symbolics. Following this upload the microprocessor enables interrupts from the Symbolics and continues to sample data from the breadboard.

8.0 Symbolics Steady State Detect

A steady state condition upload from the microprocessor results in an interrupt at the Symbolics. An communication handler reads the data and places the data in a global area of memory. A message is sent to the expert system that the steady state condition has been received. The expert reads the steady state values, performs preliminary processing and goes to sleep. Interrupts from the microprocessor are enabled. A message is output to the terminal indicating that the initial configuration has been successfully initialized and that the system is ready for fault insertion. This concludes initialization processing.

B. Fault Isolation

1.0 Fault Insertion

The user determines which faults are applicable to the current configuration by analyzing the LED display and the system architecture map on the front panel. The user inserts a fault into the steady state configuration, using the front panel of the breadboard.

2.0 Microprocessor Fault Detection

The microprocessor is currently sampling nominal conditions on the breadboard a 3 ms. intervals. When a data point is sampled outside the error tolerance specified by the error bands, the microprocessor uploads the exception values to the Symbolics. This is defined to be an unsolicited data transfer. Following an unsolicited data transfer, the microprocessor disables interrupts from the breadboard, enables interrupts from the Symbolics. If the Symbolics does not respond within 5 ms. a watchdog timer fires, triggering a routine that places the breadboard into a "safe" condition. The error band(s) violated are then in "Emergency" condition.

3.0 Symbolics Fault Detection (Unsolicited Data Transfer)

An upload by the microprocessor to the Symbolics interrupts the Symbolics. A service routine handles the interrupt and places the data into global memory. A message is sent to the expert system that an exception has occurred. This triggers the expert to read the data and to begin fault evaluation.

4.0 Symbolics requests for data (solicited request for data)

The expert may require further data from the microprocessor concerning the status of the breadboard. In this event, the Symbolics sends an interrupt to the microprocessor. The Symbolics then waits on a response VIA Rs232 XMIT from the microprocessor. The microprocessor handles the RS232 interrupt, interprets the requests, reads the data from A/D and sends an interrupt VIA RS232 XMIT to the Symbolics. On the Symbolics side an ISR handles the interrupt, places the data in a global area of memory and sends a message to the expert that the data has been received.

5.0 Symbolics requests for controls (Solicited request for control)

The expert may require controls on its behalf by the microprocessor. The sequence of events for requesting a control is the same as that described above. The control is performed by the microprocessor.

6.0 Evaluation loop until fault determination

The loop described by 3.0-5.0 continues until fault isolation/correction.

II. Microprocessor system software modular profile

A. Hierarchical decomposition

1.0 Self test

2.0 Data acquisition

2.1 Combination I/O board handler

2.2 A/D board handler

3.0 Main driver

3.1 Emergency band violation handler

3.2 Caution/warning band handler

3.3 Command interpreter

3.3.1 Request for data handler
3.3.1.1 Micro computer transmission handler
3.3.2 Request for control handler
3.3.2.1 Microcomputer relay control handler
3.3.3 Request for initial configuration handler

B. Software Module Descriptions

1.0 Module name: Self Test

a. Responsibilities

Check safe breadboard operation following power up.

b. Description

This program will be executed on power-up or when the user switches the system into self test from the from panel.

This program is interrupt driven by the CPU board timer on level 4. It is also resident in memory with the main program, but totally reconfigures the system. If the breadboard is to be re-started as aslave to the expert system, the main program must be re-entered from the top.

The program re-configures the 3 mSec clock CPU board to be 2 seconds. With every 2 second IRQ, the program closes another relay (according to a predefined configuration list) and limit checks the results (also predefined). The relays are closed in a direction from the power modules towards the loads, allowing power to be applied methodically to the system. The 2 second period allows the user time to watch the process (specifically the power output needles on the H.P. power supplies).

c. Subordinate routines

None

d. Implementation

8086 assembler

2.0 Data Acquisition

2.1 Module name: Combination I/O Board Handler

a. Responsibilities:

Service Main Control Panel

b. Description

The function of this module is to display current values to the 24 character LED and to handle faults inserted into the breadboard from the control panel. There are 4 sources of interrupts that must be handled. These sources and their lit positions to be polled are shown below.

1 ms. tier for switch debounce
(DB0)
Gas Discharge Display - RCVR Empty
(DB1)
Gas Discharge Display - XMIT Empty
(DB2)
User Repeat Function
(DB7)

This routine executes every 1 millisecond, provided no higher interrupt levels are currently being serviced. Its execution time will normally be extremely short, since the primary responsibility is to check if any of the main panel switch settings have been changed.

c. Subordinate routines

None

d. Implementation

8086 Assembler

2.2 Module name: A/D Board Handler

a. Responsibilities

Service A/D Board Conversions

The sampling Timer on the CPU board is the only source of interrupts on level 4.

b. Description

Although each board (there are 4) is capable of giving an IRQ when it has completed the conversion process, we choose to use an external timer. The rationale is as follows:

- An A/D conversion is composed of the following delay elements: channel select, analog mux delay, A/D conversion settling time, digital gain control. The last two elements are of variable delay and may not even be one on some boards. If each board supplied its own IRQ, we may become "out of sync" in our sampling.

- If we choose to consider handling power system faults which decay with time, we will need to dynamically adjust our sampling rate. The use of a single programmable timer to do this allows us this flexibility.

The timer is therefore set to approximately 3 milliseconds. At the timer IRQ, the handler starts at the top of the list of measurements and does limit checking on the fly. There is more than enough time to complete the entire list of measurements (even if the other IRQ handlers interrupt this program) prior to the next timer IRQ.

Since the boards are not individually IRQ driven, care must be taken to insure that good measurements are read, and that the loop timing is minimized. The following sequence of events should be observed:

1. Select a channel of A/D Board 1 and command the conversion to start.
2. Do the same for Boards 2, 3, 4.
3. Poll Board #1 until conversion complete and read measurement.
4. Select a new channel on Board 1 and restart conversion.
5. Limit check old Board 1 results.
6. Perform steps 3,4,5 rotating through the sequence Board 2, 3, 4, 1, 2, 3, 4, . . . etc.

The total conversion time is only on the order of 25 microseconds, so approximately 125 steps of the 8086 tMhz clock need to be stepped off. This is approximately 15 assembly level instructions with which to do limit checking. The limit checking process will require on the order of 25 assembly level instructions. Therefore, the limit checking is somewhat buried some of checking time in the conversion time. There are 4 boards which require 16 measurements each.

4 x 16 x 25 microseconds = 1.60 milliseconds, so the 3 millisecond timing loop is adequate, leaving 1.4 milliseconds for the non-buried limit checking and any communications interrupts. The 3 milliseconds represents the smallest possible sampling interval.

c. Subordinate routines

None

d. Implementation

8086 assembler

3.0 Module name: Main driver

a. Responsibilities

Initialization of breadboard to steady state following power up. Error flags detection during normal operations. Should an error be detected, it is the responsibility of the main program to activate the appropriate error handlers.

b. Description

The main program will place the breadboard in a safe state following power up. This includes opening all of the relays and placing the system in a stable configuration. Main will then wait on interrupts. Interrupts will be enabled from the breadboard, in the contest of a data acquisition cycle, on the RS232 port, in the context of experts systems requests from the microprocessor. Interrupts from the Symbolics have the higher priority. When such an interrupt is detected, control is passed to the command interpreter, described below. An interrupt from the breadboard triggers the error/caution/warning detection cycle, performed by the routines described in sections.

Main will run in one of two states. The first state is monitor steady state configuration mode. This occurs only following initial configuration. Here, main waits to detect a steady state. Upon steady state detection, a complete set of node information is uploaded to the Symbolics.

The second state is fault detection scan. Here main handles interrupts from the breadboard and passes control to subordinate routines for emergency/warning/caution detection processing.

c. Subordinate Programs:

3.1 Emergency band violation handler

3.2 Caution/warning band handler

3.3 Command Interpretor

d. Implementation

8086 assembler

3.1 Module name: Emergency band violation handler (acts as a software circuit breaker)

a. Responsibilities

Detect and flag voltage and current measurements out of range and set relays in order to ensure safe breadboard operation.

b. Description

This program is activated when a node has violated the EMERGENCY board limit for either current or voltage. The routine is responsible for determining that the measurement is not spurious, and that the error condition is stable. This routine commands the opening of the relay which controls the node(s) is the watchdog timer fires. Following a control initiated by this module, the module should send a message to the expert system that a relay has been forced open or closed.

c. Subordinate routines

Microcomputer transmission handler

d. Implementation

8086 assembler

3.2 Module name: Caution/warning band handler

a. Responsibilities

Detect and flag current and voltage out of range

b. Description

This program is activated when a caution or warning band has been violated and there exists no EMERGENCY band violation anywhere in the system. This routine is responsible for building the communication stream to be sent to the Symbolics, upon determination of an error condition.

c. Subordinate routines

Microcomputer transmission handler

d. Implementation

8086 assembler

3.3 Module name: Command Interpreter

a. Responsibilities

Control is passed to this routine upon interruption of Main by the Symbolics. This routine must handle the message, parse the command byte and pass control to the appropriate routine.

b. Description

This module is a command interpreter. Valid commands are detailed in the interface software description. Upon interrupt from the Symbolics the module places the message in an input buffer and parses the command byte. If the command is valid the starting address and ending address of the data in memory are passed to the appropriate routine. The command interpreter then returns to a read and wait state, awaiting further interrupts from the Symbolics.

c. Subordinate routines

Request for data handler

Request for control handler

Request for initial configuration handler

d. Implementation

8086 assembler

3.3.1 Module name: Request for data handler

a. Responsibilities

Retrieve and format data for transmission to the Symbolics

b. Description

Control passes to this module from the command interpreter. This routine must parse the request for data command, determine for which nodes it must gather data, read the data from memory, format the data for transmission and pass the starting and ending address of the stream to the transmission handler.

c. Subordinate routines

Microcomputer transmission handler

d. Implementation

8086 assembler

III. Symbolics system software modular profile

A. Hierarchical decomposition

1.0 Expert system command interpreter/receiver

- 1.1 Unsolicited data transfer handler
- 1.2 Solicited data transfer handler
- 1.3 Unsolicited relay transfer handler
- 1.4 Steady state values handler

2.0 Expert system communication sender

- 2.1 Request for data from microprocessor
- 2.2 Request for control by microprocessor
- 2.3 Request for configuration download

B. Software module descriptions

1.0 Module name: Expert system command interpreter receiver (ESCR)

a. Responsibilities

Handle transmissions from microprocessor

b. Description

The ESCR will handle interrupts from the microprocessor, validate the input stream, store the data in memory, interpret the command byte of the header record and, based on the value of the command byte, transfer control to the appropriate subordinate function.

c. Subordinate functions

Unsolicited data transfer handler

Solicited data transfer handler

Unsolicited relay transfer handler

Steady state values handler

d. Implementation

Lisp machine process

1.1 Module name: Unsolicited data transfer handler

a. Responsibilities

Interface between ESCR and expert system

b. Description

When ESCR detects and unsolicited data transfer, control is transferred to this function. The function processes the data, formats it for compatibility with the expert system, transfers the data into memory and notifies the expert system that the transfer has occurred.

c. Subordinate routines

Expert system

d. Implementation

Lisp function, called from ESCR.

1.2 Module name: Solicited data transfer handler

a. Responsibilities

Interface between ESCR and expert system

b. Description

When ESCR detects an solicited data transfer, control is transferred to this function. The function processes the data, formats it for compatibility with the expert system, transfers the data into memory and notifies the expert system that the transfer has occurred.

c. Subordinate routines

Expert system.

d. Implementation

Lisp function, called from ESCR.

1.3 Module name: Unsolicited relay transfer handler

a. Responsibilities

Interface between ESCR and expert system

b. Description

When ESCR detects an unsolicited relay transfer, control is transferred to this function. The function processes the data, formats it for compatibility with the expert system, transfers the data into memory and notifies the expert system that the transfer has occurred.

c. Subordinate routines

Expert system

d. Implementation

Lisp function, called from ESCR.

1.4 Module name: Steady state values handler

a. Responsibilities

Interface between ESCR and expert system

b. Description

When ESCR detects and steady state value transfer, control is transferred to this function. The function processes the data, formats it for compatibility with the expert system, transfers the data into memory and notifies the expert system that the transfer has occurred.

c. Subordinate routines

Expert system

d. Implementation

Lisp function, called from ESCR.

2.0 Module name: Expert system communication sender (ESCS)

a. Responsibilities

Send transmissions to microprocessor

b. Description

The ESCS will download data streams to the microprocessor.

c. Subordinate functions

None

d. Implementation

Lisp function, called by routines sending data to the microprocessor.

2.1 Module name: Request for data from microprocessor

a. Responsibilities

Format requests for data from expert system to microprocessor

b. Description

This function will be called by the expert system. The expert system will specify the data items for which it requests information. This routine will format the request, include the transmission header information and pass control to the ESCS for transmission of the output stream.

c. Subordinate functions

ESCS

d. Implementation

Lisp function, called from expert system.

2.2 Module name: Request for control by microprocessor

a. Responsibilities

Format requests for controls from expert system to microprocessor

b. Description

This function will be called by the expert system. The expert system will specify the relays for which it requests controls. This routine will format the request, include the transmission header information and pass control to the ESCS for transmission of the output stream.

c. Subordinate functions

ESCS

d. Implementation

Lisp function, called from expert system.

2.3 Module name: Request for configuration download

a. Responsibilities

Format requests for initial configuration by microprocessor

b. Description

This function will be called from the user interface. The function will be passed the name of the file containing the initial configuration for the breadboard. This includes relay-node mappings, current-voltage mappings and initial warning and caution error bands. This routine will format the request, include the transmission header information and pass control to the ESCS for transmission of the output stream.

c. Subordinate functions

ESCS

d. Implementation

Lisp function, called from user interface.

3.3.1.1 Module name: Microcomputer Transmission Handler

a. Responsibilities

Send formatted data streams to the Symbolics

b. Description

This module called as a subroutine to the main program, and other service routines needing to transmit data to the expert system. The message to be transmitted will be placed in an input buffer somewhere in the memory of the microcomputer. The starting and ending addresses of the stream to be transmitted will be passed to the transmission handler. The program will check for valid protocol and upload the stream to the Symbolics.

c. Subordinate routines

None

d. Implementation

8086 assembler

3.3.2 Module name: Request for control handler

a. Responsibilities

Perform controls on behalf of Symbolics

b. Description

Control passes to this module from the command interpreter. This routine must parse the request for controls(s) command, determine for which relays it must execute controls and pass the relay numbers to the microcomputer control handler.

c. Subordinate routines

Microcomputer relay control handler

d. Implementation

8086 assembler

3.3.2.1 Module name: Microcomputer Relay Control Handler

a. Responsibilities

Control Relays

b. Description

This program is non-IRQ driven and is called as a subroutine to the main program, and other service routines needing to alter the relay settings.

The board requires a 48 bit mask register to enable/disable the system relays. The board also has an override switch which disables all relays in the event of system power up or emergencies.

The board handler processes a list of relay numbers which are to be turned on. The handler constructs the 48 bit mask to the board in 16 bit quantities.

c. Subordinate routines

None

d. Implementation

8086 assembler

3.3.3 Module name: Request for initial configuration handler

a. Responsibilities

Perform initialization per instructions contained in configuration download message from Symbolics.

b. Description

Control passes to this module from the command interpreter. This routine must initialize the breadboard to the requested state and initialize data tables in microcomputer memory. These tables include error band values and node-relay mappings. This program will return control to Main and set the monitor initial steady state flag in Main.

c. Subordinate routines

Microcomputer relay control handler

d. Implementation

8086 assembler

IV. Symbolics application expert system software modular profile

A. Hierarchical decomposition

1.0 User interface

2.0 Configuration initialization

3.0 Fault-data initialization

4.0 Hypothesis Generation

5.0 Hypothesis mediation

6.0 Correction

7.0 Explanation

B. Software module descriptions

1.0 Module name: User interface

a. Responsibilities

Allow user to select an existing configuration or set up an initial configuration

b. Description

This program will be implemented in Lisp or ART. It will access a small data base of existing configurations. It will also allow for user defined configurations. Once a configuration has been defined, it will be downloaded to the microsystem. Ideally, this will be an interactive module that will incorporate the ART graphics package.

c. Subordinate functions

Request for configuration download.

d. Implementation

ART and Lisp on the Symbolics.

2.0 Module name: Configuration initialization

a. Responsibilities

Establish the system configuration description and the error band data within the ART environment.

b. Description

This program will be implemented in ART. The circuit components and nodes will be predefined. The operations to be implemented will be: (1) determination of configuration dependent source/sink relationships; and (2) insertion of configuration dependent error bands.

c. Subordinate functions

Notice of completion sent to interface/user. System ready to insert fault.

d. Implementation

ART module.

3.0 Module name: Fault-data initialization

a. Responsibilities

Enter data incurred after a fault was inserted.

b. Description

This program will be implemented in ART. The main function will be to enter faulted data into the ART representation of the system.

c. Subordinate functions

None.

d. Implementation

ART module

4.0 Module name: Hypothesis generation

a. Responsibilities

Generate hypotheses including: fault type and fault location

b. Description

This module will be implemented in ART. It will be based on: (1) qualitative reasoning; (2) source/sink relationships; (3) fault descriptions and (4) a modular representation of the power system.

c. Subordinate functions

None

d. Implementation

ART module.

5.0 Module name: Hypothesis mediation

a. Responsibilities

Prune off incorrect hypotheses.

b. Description

This module will be implemented using ART;s viewpoint facility. The techniques used will be both causal and hieristic-based. In specific cases, directives to flip relays will be sent to the breadboard.

c. Subordinate functions

Directives to change the relay configuration and return the effected data.

d. Implementation

ART module.

6.0 Module name: Correction

a. Responsibilities

If possible, reconfigure the system to work around the fault. Otherwise make proper recommendations to the user.

b. Description

This module will be implemented in ART. It will make use of heuristic and causal reasoning techniques when applicable. It will send relay change directives to the breadboard when necessary. It will hypothesize a correction, analyze it, and request it.

c. Subordinate functions

Relay change directives will be sent to the breadboard.

d. Implementation

ART module.

7.0 Module name: Explanation

a. Responsibilities

Report fault and explanation to the user.

b. Description

Report the fault and a trace of the reasoning path followed to arrive at the decision. This may allow for limited user interaction, and may incorporate graphics into the explanation facility.

c. Subordinate functions

None

d. Implementation

ART module.

V. Interface description

A. Overview

In this section we define the interfaces in term of sender-receiver relationships in the system. There are 4 sender relationships and 4 receiver relationships. Sender relationships are breadboard to microprocessor, microprocessor to Symbolics, Symbolics to microprocessor and microprocessor to breadboard. Receivers are microprocessor from breadboard, Symbolics from microprocessor, microprocessor from Symbolics and breadboard from microprocessor.

Examination of these sender-receiver relationships allows a framework for defining the exact data that must be transferred across any given interface and also allow the identification of the software modules that must be constructed in the interface software.

B. Transmission senders

1.0 Breadboard to microprocessor

1.1 Requirement: Data generation

The breadboard will generate complete data sets at a 3ms. frequency. Data acquisition by the microprocessor will be interrupt driven. The microprocessor must handle interrupts, perform limit checks and store the data within the 3 ms. time frame.

2.0 Microprocessor to Symbolics

2.1 Requirement: Unsolicited data transfers

The microprocessor must detect and report exceptions relative to the initial configuration error bands. This detection must occur within the 3 ms. data acquisition cycle described above.

2.2 Requirement: Solicited data transfers

The microprocessor must send data to the Symbolics following a request for data.

2.3 Requirement: Report unsolicited forced controls

The microprocessor must send data to the Symbolics following a forced control. A forced control is initiated by the microprocessor in order to prevent board failure. This will occur only under extraordinary circumstances. In this respect the microprocessor acts as a software circuit breaker.

2.4 Requirement: Steady state data value upload

The microprocessor will send the steady state data values to the Symbolics following configuration load and steady state detect.

3.0 Symbolics to microprocessor

3.1 Requirement: Configuration load

The Symbolics must send the initial configuration to the microprocessor.

3.2 Requirement: Solicited requests for data

The Symbolics will send requests for data to the microprocessor.

3.3 Requirement: Solicited request for control

The Symbolics will send requests for controls to the microprocessor.

4.0 Microprocessor to breadboard

4.1 Requirement: Set safe configuration

The microprocessor must initialize the breadboard to a safe configuration following power on self test.

4.2 Requirement: Set initial configuration

The microprocessor must initialize the breadboard to the initial configuration requested by the user.

4.3 Requirement: Set relays on request for control

The microprocessor must respond to requests for control from the expert.

C. Transmission receivers

1.0 Microprocessor from breadboard

1.1 Requirement: Data Acquisition

The microprocessor must be capable of handling interrupt driven data acquisition at the rate of 3ms. per cycle.

2.0 Symbolics from microprocessor

2.1 Requirement: Handle unsolicited data transfers

The Symbolics must handle interrupts from the microprocessor, identify unsolicited data transfers and pass the data to the expert system.

2.2 Requirement: Handle solicited data transfers

The Symbolics must handle interrupts from the microprocessor, identify solicited data transfers and pass the data to the expert system.

2.3 Requirement: Handle unsolicited control report

The Symbolics must handle interrupts from the microprocessor, identify unsolicited control reports and pass the data to the expert system.

2.4 Requirement: Handle steady state value table upload

The Symbolics must handle interrupts from the microprocessor, identify steady state table uploads and pass the data to the expert system.

3.0 Microprocessor from Symbolics

3.1 Requirement: Handle request for configuration load

The microprocessor must handle interrupts from the Symbolics, identify request for a configuration load and transfer control to the routine that performs configuration load.

3.2 Requirement: Handle request for data (solicited)

The microprocessor must handle interrupts from the Symbolics, identify a request for data and transfer control to the return that performs solicited request for data.

3.3 Requirement: Handle request for control (solicited)

The microprocessor must handle interrupts from the Symbolics, identify a request for control and transfer control to the routine that handles solicited request for control.

4.0 Breadboard from microprocessor

4.1 Requirement: Set relays

The breadboard must respond to configuration set commands issued by the microprocessor.

D. Data transmission formats

1.0 Protocol description

1.1 Header records

This section describes header records for data transmission formats. These header records will be used to identify the type of message that is being sent and to trigger the appropriate receiver function on the receiver side of the interface. Record formats are described by relative byte address in the record.

1.2 Header record format

Command byte	0
Checksum	1-2

1.3 Trailer records

Trailer records are used to indicate end of transmission.

1.4 Trailer record format

End of transmission

0

Always '*'

2.0 Transmission formats

A transmission, including headers, trailers and the description of the transmitted data is described for each of the messages in the system.

2.1 Data generation by breadboard

2.2 Microprocessor to symbolics

2.2.1 Unsolicited data transfer

Command byte	0	always = 'u'
Checksum	1-2	16 bit integer
Node id	3-4	16 bit integer
Error type	5	
' ' = N/A		
'n' =		
'c' = caution		
'w' = warning		
'e' = error		
Error direction	6	
' ' = N/A		
'h' = above limit		
'l' = below limit		
Voltage	7-12	
Current	13-19	

Repeating groups, bytes 3-19 until trailer record is encountered (see trailer record, end of transmission).

2.2.2 Solicited data transfer

Command byte	0	always 's'

(same as 3.2.1.1)

2.2.3 Unsolicited relay status: relay forced closed

Command byte	0	always 'r'
Relay id	1-2	
Relay status	3	
'o' = open		
'c' = closed		
'f' = forced close		

Repeating groups, bytes 1-3 until trailer record is encountered (see trailer record, end of transmission).

2.2.4 Steady state table upload

Command byte	0	always 'z'
(same as 3.2.1.1)		

2.3 Symbolics to microprocessor

2.3.1 Configuration load

Command byte	0	always 'i'
Checksum	1-2	16 bit integer
Node id	3-4	16 bit integer
Voltage flag	5	
1 = measure voltage		
0 = don't		
Current flag	6	
Error band low	7-12	
Warning band low	13-18	
Caution band low	19-24	
Error band high	25-30	
Warning band high	31-36	
Caution band high	37-42	

Repeating group of bytes 3-42 until end of transmission

2.3.2 Requests for data transfer (solicited)

Command byte	0	always 'd'
Node id	1-2	

Repeating groups of bytes 1-2 until end of transmission

2.3.3 Requests for control

Command byte	0	always 'c'
Relay id	1-2	
Relay status	3	
'o' = open		
'c' = closed		
'f' = force close		

Repeating groups, bytes 1-3 until trailer record is encountered (see trailer record, end of transmission).

2.3.4 Microprocessor to breadboard

2.3.4.1 Set relays

E. Command function summary

1.0 Microprocessor to symbolics

'u' returning unsolicited data

's' returning solicited data

'r' returning unsolicited relay status

'z' returning steady state value table

2.0 Symbolics to microprocessor

'i' configuration load

'd' request for data

'c' request for control

APPENDIX D

FIES Expert System Rules with Translation

ERRORBANDS

```
DefRule Create-Errorbands-Zero-current-Zero-voltage
        (declare   (salience 850))
    (sensor-status-num-c ?sensor ?current)
    (sensor-status-num-v ?sensor /voltage)
    (test (and (< ?current .15) (< ?voltage 1.3)))

    (Modify
         (Schema ?sensor
              (normal-current ?current)
              (normal-voltage ?voltage)
              (current-errorband  (-1.0 * * .15))
              (voltage-errorband  (-1 * * 1.3 )))))
"For a given sensor, if its normal current is less than .15 and its voltage is
    less than 1.3 then its current errorband is -1.0 to .15 and its voltage
    errorband is from -1 to 1.3."

(DefRule Create-Errorbands-Zero-Current
    (declare  (salience 850))
    (sensor-status-num-c ?sensor ?current)
    (sensor-status-num-v ?sensor ?voltage)
    (test (and (< ?current .15) (not (< ?voltage 1.3))))

    (Modify
         (Schema ?sensor
              (normal-current ?current)
              (normal-voltage ?voltage)
              (current-errorband  (-1.0 * * .15 ))
              (voltage-errorband  (-1 1.3 +(?voltage - ?voltage * .1) =(?voltage + ?voltage * .1))))))
```

```
(DefRule Create-Errorbands-Less-Than-AMP
        (declare   (salience 850))
    (sensor-status-num-c ?sensor ?current)
    (sensor-status-num-v ?sensor ?voltage)
    (test (and (not (< ?current .04))(< ?current 1.0) (not (< ?voltage 1.3))))

    (Modify
         (Schema ?sensor
              (normal-current ?current)
              (normal-voltage ?voltage)
              (current-errorband  (-1.0 .04  +(?current - .1) =(?current + .1)))
              (voltage-errorband  (-1 1.3 +(?voltage - ?voltage * .1) =(?voltage + ?voltage * .1))))))
(DefRule Create-Errorbands
    (declare  (salience 850))
    (sensor-status-num-c ?sensor ?current)
    (sensor-status-num-v ?sensor ?voltage)
    (test (and (not (< ?current 1.0)) (not (< ?voltage 1.3))))

   (Modify
        (Schema ?sensor
             (normal-current ?current)
             (normal-voltage ?voltage)
             (current-errorband  (-1.0 .15 =(?current - ?current * .1) =(?current = ?current * .1)))
             (voltage-errorband  (-1 1.3 =(?voltage - ?voltage * .1) =(?voltage + ?voltage * .1))))))
```

P-DESCRIBE-CIRCUIT-2

```
Rule Power-Source-to-Relay-Current-Flow
    (declare (salience 1000))
    (Schema ?power-source
        (Instance-of Power-Source)
?fl     (Potential-Sink ?relay))

    Schema ?relay
        (Instance-of Relay)
        (Status closed))

    (Assert
        Schema ?power-source
            (sink ?relay)))
    (Assert
        (Schema ?relay
            (source ?power-source)))
    (Retract ?fl))
```

"If some power source has some relay as a potential sink and the relay is closed then the relay is the sink for the power source."

```
(DefRule Component-to-Component-Current-Flow
        (declare    (salience 1000))
    Schema ?source-component
        (Instance-of Component)
        (Source ?)
?fl     (Potential-Sink ?sink-component))
    (Schema ?sink-component
        (Instance-of Component:Power-Sink))
    (not
        (Schema ?sink-component
            (Instance-of Relay))

    (Assert
        (Schema ?source-component
            (sink ?sink-component)))

    (Assert
        (Schema ?sink-component
            (source ?source-component)))
    (Retract ?fl))
```

"If some source component is a component of the network and its sink is its potential sink component is in the network then the potential sink is in fact a sink for the source."

D-4

```
(DefRule Component-to-Relay-Current-Flow
        (declare    (salience 1000))
    (Schema ?source-component
        (Instance-of Component)
        (Source ?)
?fl     (Potential-Sink ?sink-component))
    (Schema ?sink-component
        (Instance-of Relay)
        (Status Closed))

    (Exists
        (Schema ?source-component
            (Source ?)))

    (Assert
        (Schema ?source-component
            (sink ?sink-component)))

    (Assert
        (Schema ?sink-component
            (source ?source-component)))
    (Retract ?fl))
```

"If a source has both a relay and a nonrelay sink, consider the nonrelay sink to be the sink."

ORIGINAL PAGE IS
OF POOR QUALITY

```
(DefRule Power-Source-to-Component-Source-Sensor
        (declare    (salience 990))
    (Schema ?power-source)
        (Instance-of Power-Source)
        (Sink ?sink-component))

    (Schema ?sink-component
        (instance-of Component))



    (Assert
        (Schema ?sink-component
            (Source-Sensor none))))
```

"If the source for a sink is a power source, then the sink has no source sensor."

```
(DefRule Component-to-Component-Source-Sensor
        (declare    (salience 980))
    (Schema ?source-component
        (Instance-of Component)
        (Source-Sensor ?source-sensor)
        (Sink ?sink-component))
    (Schema ?sink-component
        (Instance-of Component))
    (not
        (Schema ?sink-component
            (Instance-of Sensor:Power-Sink)))

    (Assert
        (Schema ?sink-component
            (source-sensor ?source-sensor))))
```

"For a given source, if it's sink is not a sensor or a power sink then the source sensor for the sink is the source's sensor."

```
(DefRule Component-to-Sensor-to-Component-Sink-Sensor-Source-Sensor
    (declare    (salience 970))
    (Schema ?source-component
        (Instance-of Component)
        (Source-Sensor ?)
        (Sink ?sensor))
    (Schema ?sensor
        (Instance-of Sensor)
        (Sink ?sink-component))
    (Schema ?sink-component
        (Instance-of Component))

    (Assert
        (Schema ?source-component
            @sink-sensor ?sensor)))

    (Assert
        (Schema ?sink-component
            (source-sensor ?sensor))))
```

"This rule is assuring that the schemas for the sensors for sources and sinks in fact say that they are the sensors for the sources and sinks."

```
(DefRule Component-toPower-Sink-Source-Sensor-Sink-Sensor
    (declare    (salience 970))
    (Schema ?source-comopnent
        (Instance-of Component)
        (Source-Sensor ?source-sensor)
        (Sink ?power-sink))
    (Schema ?power-sink
        (Instance-of Power-Sink))

    (Assert
        (Schema ?source-component
            (Sink-Sensor none)))

    (Assert
        (Schema ?power-sink
            (source-sensor ?source-sensor))))

"If the sink for a given source is a power sink, then the source sensor is
    the sensor for the source but there is no sink sensor."
```

```
(DefRule Component-to-Component-Sink-Sensor
        (declare    (salience 960))
    (Schema ?sink-component
        (Instance-of Component
        (Sink-Sensor ?sink-sensor)
        (Source ?source-component))
    (not
        (Schema ?sink-component
            (Instance-of Power-Sink)))
    (Schema ?source-component
        (Instance-of Component|Power-Source))
    (not
        (Schema ?source-component
            (Instance-of Sensor)))

    (Assert
        (Schema ?source-component
            (sink-sensor ?sink-sensor))))
```

"If the source for a component is not a sensor and the sink is not a power sink then the sink sensor is the sensor for the sink component."

```
(DefRule Assign-I-RQ-source

    (declare    (salience 950))
    (Schema ?component
        (Instance-of Component)
        (I-Rq  (    (+ $?source-I ) (+ $?sink-I)))
        (Source ?source))
    (not
        (Schema ?component
            (I-Rq ((+ $? ?source $?) ?))))

    (Modify
        (Schema ?component
            (I-Rq ((+ $?source-I ?source) (+ $?sink-I))))))
```

"This rule builds a list of sources for a component and stores the list in I-RQ."

```
(DefRule Assign-I-RQ-sink

    (declare    (salience 940))
    (Schema ?component
        (Instance-of Component)
        (I-Rq  (    ( $?source-i)   (+ $?sink-i)))
        (Sink ?sink))
    (not
        (Schema ?component
            (I-Rq (?    (+ $? ?sink $?)))))

    (Modify
        (Schema ?component
        (I-Rq  (    (+ $?source-1) (+$?sink-1 ?sink))))))

"This rule builds a list of sinks."


(DefRule Assign-Multiple-Sources
    (Declare    (salience 930))
    (Schema ?bus
        (Instance-of bus)
        (Multiple-Sources no)
        (Source ?1)
        (Source ?2&~?1))

    (Modify
        (Schema ?bus
            (Multiple-Sources yes))))

"If the schema for a bus indicates that it does not have multiple sources, but
    in fact it has multiple sources; then  modify the schema to show that it
    does have multiple sources."
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
(DefRule Assign-Multiple-Sinks
    (Declare    (salience 910))
    (Schema ?bus
        (Instance-of bus)
        (Multiple-Sinks no)
        (Sink ?1)
        (Sink ?2&~?1))

    (Modify
        (Schema ?bus
            (multiple-Sinks yes))))

"Same as above for sinks."

(DefRule Assign-Coupled
    (Declare    (Salience 920))
    (Schema ?bus
        (Instance-of bus)
        (Multiple-Source yes)
        (Source-Sensor ?source-sensor))
    (Schema ?source-sensor
        (Instance-of Sensor)
        (Coupled-Sink no))


    (Modify
        (Schema ?source-sensor
            (Coupled-Sink yes))))

"If a bus has multiple sources, modify the schema of the source sensor for the
    to show that the source has a coupled sink."
```

```
(DefRule Find-Potential-Relay-Sinks
    (Declare    (salience 905))
    (Schema ?component
        (Source-Sensor ?source-sensor)
        (potential-Sink ?potential-sink))

    (Schema ?opotential-sink
        (Instance-of Relay)
        (Status Open)
        (Potential-Sink ?load))

    (Schema ?load
        (Instance-of load))

    (Assert
        (Schema ?source-sensor
            (Potential-Relay-Sink ?potential-sink))))
```

"If the potential sink for a source is an open relay and its potential sink is a load then assert that the relay is a potential sink."

ORIGINAL PAGE IS
OF POOR QUALITY

```
(DefRule Explain-Hypothesis-Gen
    (declare (salience 671))

    (printout t t " Hypothesizing Faults: " )
    (printout t t " ")
    (printout t t " "))

(DefRule Find-Open-Circuit
    (declare (salience 670))
    (Schema ?component
         (instance-of component)
         (source-sensor ?source-sensor)
         (sink-sensor ?sink-sensor))

    (Schema ?source-sensor
         (instance-of sensor)
         (Qual-Current Zero:Low)
         (Qual-Voltage Normal:High ))

    (Schema ?sink-sensor
         (instance-of sensor)
         (Qual-Current Zero)
         (Qual-Voltage Zero))

    (Viewpoint ?vp
         (nor (Fault-Hypothesis Open-Circuit ?component)))


    (Sprout   (Assert (Fault-Hypothesis Open-Circuit ?component))))
```

"If the source sensor for a component is registering zero or low current and normal or high voltage and its sink sensor is registering zero current and voltage then hypothesize that there is an open circut at the component."

```
(DefRule Finkd-Open-Circuit-nos-source
    (declare  (Salience 670))
    (Schema ?component
         (instance-of power-source)
         (source-sensor none)
         (sink-sensor ?sink-sensor))


    (Schema ?sink-sensor
         (instance-of sensor)
         (Coupled-Sink no)
         (Qual-Current Zero)
         (Qual-Voltage Zero))

    (Viewpoint      ?vp
         (not (Fault-Hypothesis Open-Circuit ?component)))
    (Sprout   (Assert (Fault-Hypothesis Open-Circuit ?component))))
```

" If a given component has no source, its sink is not coupled and the sink sensor volatage and current are zero then hypothesize an open circut at the component."

```
(DefRule Find-Open-Circuit-no-sink

    (declare (salience 670))
    (Schema ?component
         (instance-of component:power-sink)
         (source-snesor ?source-sensor)
         (sink-sensor none))


    (Schema ?source-sensor
         (instance-of sensor)
         (Qual-Current Zero:Low)
         (Qual-Voltage Normal:high ))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Open-Circuit ?component)))

    (Sprout (Assert (Fault-Hypothesis Open-Circuit ?component))))
```

"If a component is a power sink with no sink sensor and its source sensor shows zero or low current and normal or high voltage then hypothesis an open circuit for the component."

```
(DefRule Find-Open-Circuit-Cold-Sk
    (declare (salience 670))
    (Schema ?component
         (instance of component)
         (Source-Sensor ?source-sensor)
         (Sink-Sensor ?sink-sensor))

    (Schema ?csource-sensor
         (instance-of sensor)
         (Qual-Current Zero)
         (Qual-Voltage Normal))

    (Schema ?sink-sensor
         (instance-of sensor)
         (Qual-Current Zero)
         (Qual-Voltage Low:Normal)
         (Coupled-Sink yes))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Open-Circuit ?component)))
    (Sprout (Assert (Fault-Hypothesis Open-Circuit ?component))))
```

If a component's source sensor registers zero current and normal voltage and its sink is a coupled sink whose sensor registers zero current and low or normal voltage then hypothesize an open circuit for the component."

```
(DefRule Find-Open-Circuit-Cold-no-source
    (declare (salience 670))
    (Schema ?component
         (instance-of power-source)
         (not (instance-of relay))
         (Source-Sensor none)
         (Sink-Sensor ?sink-sensor)

    (Schema ?sink-sensor
         (instance-of sensor)
         (Qual-Current Zero)
         (Qual-Voltage Low|Normal)
         (Coupled-Sink yes))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Open-Circuit ?component)))


    (Sprout (Assert (Fault-Hypothesis Open-Circuit ?component))))
```

If a source is a power source with no sensor--but not a relay--and its sink is a coupled sink whose sensors register zero current and low or no voltage then hypothesize an open circuit for the component."

ORIGINAL PAGE IS
OF POOR QUALITY

```
(DefRule Find-Short-Circuit
    (declare (salience 670))
    (Schema ?component
         (instance-of component)
         (not (instance-of relay))
         (source-sensor ?source-sensor)
         (sink-sensor ?sink-sensor))

    (Schema ?source-sensor
         (instance-of sensor)
         (Qual Current High:Sys-Limit)
         (Qual-Voltage Low:Zero ))

    (Schema ?sink-sensor
         (instance-of sensor)
         (Qual-Current Zero)
         (Qual-Voltage Zero))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Short-Circuit ?component)))

    (Sprout (Assert (Fault-Hypothesis Short-Circuit ?component))))
```

"If a component is not a relay and its source sensor registers high or system limited current and low or zero voltage and its sink sensor registers zero current and voltage then hypothesize a short circuit."

ORIGINAL PAGE IS
OF POOR QUALITY

```
(DefRule Find-Short-Circuit-cold
    (declare (salience 670))
    (Schema ?comonent
         (instance-of component)
         (not (instance-of relay))
         (source-sensor ?source-sensor)
         (sink-sensor ?sink-sensor))


    (Schema ?source-sensor
         (instance-of sensor)
         (Qual-Current High:Sys-Limit)
         (Qual-Voltage Low:Zero ))

    (Schema ?sink-sensor
         (instance-of sensor)
         (Coupled-Sink yes)
         (Qual-Current Zero)
         (Qual-Voltage Low:Zero:Normal))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Short-Circuit ?component)))

    (Sprout (Assert (Fault-Hypothesis Short-Circuit ?component))))
```

"If a component is not a relay and its source sensor registers high or system limited current and low or zero voltage and its sink is a coupled sink whose sensor registers zero current and low, zero or normal voltage then hypothesize a short circuit."

```
(DefRule Find-Short-Circuit-no-source
    (declare (salience 670))
    (Schema ?component
         (instance-of power-source)
         (not (instance-of relay))
         (source-sensor none)
         (sink-sensor ?sink-sensor))


    (Schema ?sink-sensor
         (instance-of sensor)
         (Coupled-Sink no)
         (Qual-Current Zero)
         (Qual-Voltage Low;Zero))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Short-Circuit ?component)))

    (Sprout (Assert (Fault-Hypothesis Short-Circuit ?comopnent))))
```

"If a component is a power source whose sink is not a coupled sink and the sink sensor registers zero current and low or zero voltage then hypothesize a short circuit."

```
(DefRule Find-Short-Circuit-Cpled-no-source
    (declare (salience670))
    (Schema ?component
         (instance-of power-source)
         (not (instance-of relay))
         (source-sensor None)
         (sink-sensor ?sink-sensor))

    (Schema ?sink-sensor
         (instance-of sensor)
         (Coupled-Sink yes)
         (Qual-Current Zero)
         (Qual-Voltage Low:Zero))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Short-Circuit ?component)))

    (Sprout (Assert (Fault-Hypothesis Short-Circuit ?component))))

"If a power source has a coupled sink whose sensor registers zero current and
    low or zero voltage, then hypothesize a short circuit."
```

```
(DefRule Find-Short-Circuit-nosink
    (declare (salience 670))
    (Schema ?component
         (instance-of power-sink:component)
         (not (instance-of relay))
         (source-sensor ?source sensor)
         (sink-sensor none))

    (Schema ?source/sensor
         (instance-of sensor)
         (Qual-Current High:Sys-Limit)
         (Qual-Voltage Low:Zero ))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Short-Circuit ?component)))


    (Sprout (Assert (Fault-Hypothesis Short Circuit ?component))))

"If a component is a power sink and its source sensor registers high or system
    limited current and low or zero voltage then hypothesize a short circuit."
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
(DefRule Find-Resistive-Short-Circuit
    (declare (salience 670))
    (Schema ?component
         (instance-of component)
         (not (instance-of relay))
         (source sensor ?source sensor)
         (sink-sensor ?sink-sensor))

    (Schema ?sourse-sensor
         (instance-of sensor)
         (Qual-Current High:Sys-Limit)
         (Qual-Voltage Low:Normal ))

    (Schema ?sink-sensor
         (instance-of sensor)
         (Qual-Current Low:Normal)
         (Qual-Voltage Low:Normal))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Resistive-Short-Circuit ?component)))

    (Sprout (Assert (Fault-Hypothesis Resistive-Short-Circuit ?component))))
```

"If a component is not a relay and source current is high or system limited, source voltage is low or normal and sink current and voltage is low or normal then hypothesize a short circuit."

ORIGINAL PAGE IS
OF POOR QUALITY

```
(DefRule Find-Resistive-Short-Circuit-no-source
    (Declare (salience 670))
    (Schema ?component
         (instance-of power-source)
         (not (instance-of relay))
         (source-sensor none)
         (sink-sensor ?sink-sensor))


    (Schema ?sink-sensor
         (instance-of sensor)
         (Coupled-Sink no)
         (Qual-Current Zero:Low)
         (Qual-Voltage Zero:Low:Normal))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Resistive-Short-Circuit ?component)))

    (Sprout (Assert (Fault-Hypothesis Resistive-Short-Circuit ?component))))
```

"If the sink sensor for a power source registers zero or low current and zero, low or normal voltage and the sink is not coupled then hypothesize a resistive short circuit."

```
(DefRule Find-Resistive-Short-Circuit-cpld-no-source
    (declare (salience 670))
    (Schema ?component
         (instance-of power-source)
         (not (instance-of relay))
         (source-sensor none)
         (sink-sensor ?sink-sensor))

    (Schema ?sink-sensor
         (instance-of sensor)
         (Coupled-Sink yes)
         (Qual-Current Zero:Low)
         (Qual-Voltage Low:Normal))


    (Viewpoint ?vp
         (not (Fault-Hypothesis Resistive-Short-Circuit ?component)))

    (Sprout (Assert (Fault-Hypothesis Resistive-Short-Circuit ?component))))
```

"If the sink for a power source is coupled and the sink's sensor registers zero or low current and low or normal voltage then hypothesize a short circuit."

```
(DefRule Fink-Closed-Relay-no-sksen
    (declare (salience 670))
    (Schema ?source-sensor
         (instance-of sensor)
         (Qual-Current Hight)
         (Qual-Voltage Normal;Low)
         (Potential-Relay-Sink ?relay))

    (Viewpoint ?vp
         (not (Fault-Hypothesis Closed-Relay ?relay)))

    (Sprout (Assert (Fault-Hypothesis Closed-Relay ?relay))))
```

"If a source sensor has no corresponding sink sensor because the sink is a relay and the source sensor registers high current and normal or low voltage then hypothesize a closed relay."

ORIGINAL PAGE IS
OF POOR QUALITY

```
(DefRule Explain-Read-In
        (declare  (salience 582))

        (printout t t " Reading In Faulted Values " )
        (printout t t " "))

(DefRule Read-In-Faulted-Values
    (declare  (salience 750))
    (sensor-status ?sensor ?current ?voltage)

    (Modify
        (Schema ?sensor
            (faulted-current ?current)
            (faulted-voltage ?voltage))))
```

"This rule modifies the database to contain the faulted values, setting the current and voltage for each sensor."

```
(DefRule Determine-Qualitative-Values-Current
    (declare  (salience 740))

    (Schema ?sensor
        (faulted-current ?current)
        (current-errorband (?neg ?zero ?low ?high)))

    (bind ?qual-value  (assign-qual ?current ?neg ?zero ?low ?high))
    (Modify
        (Schema ?sensor
            (qual-current ?qual-value))))
```

"This rule assign the qualitative value (zero, low, high, etc.)to the quantitative current for each sensor based on the contents of the error band database."

ORIGINAL PAGE IS
OF POOR QUALITY

```
(DefRule Determine-Qualitative-Values-Voltage
    (declare  (salience 740))

    (Schema ?sensor
         (faulted-voltage ?voltage)
         (voltage-errorband (?neg ?zero ?low ?high)))
    (bind ?qual-value   (assign qual ?voltage ?neg ?zero ?low ?high))
    (Modify
         (Schema ?sensor
              (qual voltage  ?qual value))))

"Same as above except for voltage."


(DefRule Create graphics
    (declare  (salience 585))

         (printout t t "Creating Window")
    #L   (create-window 'fies-2 ' :graphics 10 300 1000 725)
    #L   (label-window (window stream 'fies-2) "FIES")
    #L   (expose-window 'fies-2)
    #L   (show icon 'icon-2 'fies 2)
         (printout t t " ");


(DefRule Flash All Icons
    (declare  (salience 50))
    (Schema ?x
         (Has-Icon ?icon))

    (printout t t "Flashing Icon" ?icon)
    #L   (flash-icon ?icon 3 60 'fies-2);

"These rules manipulate the FIES II command windows."
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
(Defrule Printout-All-Faults
    (declare (salience 355))

    (printout t t "Reporting All Initial Fault Hypotheses: ")
    (printout t t " " )
    (printout t t " " ))

"These rules cause output to go to the screen.  This rule simply outputs the
    heading 'Reporting All Initial Fault Hypotheses.'"


(Defrule Summarize-All Faults
    (declare (salience 350))
    (Fault-Hypothesis  ?fault ?location)

    (printout t t "      " ?fault " " ?location)
    (printout t t " "))

"This rule outputs the fault that is currently instantiated to the variable
    ?fault and the location (of the fault) which is instantiated to ?location."

(Defrule Printout-Remaining-Faults flag
    (declare (salience 345))

    (printout t t " ")
    (printout t t "Reporting Final Fault Hypotheses: ")
    (printout t t " ")
    (printout t t " "))


(Defrule Printout-Remaining-Faults
    (declare (salience 340))
    (Fault-Hypothesis ?fault ?location)
    (not (inconsistent Hypothesis ?fault ?location))
    (not (Inconsistent Hypothesis v ?fault ?location))

    (printout t t "      " ?fault " " ?location)
    (printout t t " "))
```

```
Defrule Inconsistent-Load-Short-Circuit
    (declare (salience 610))
    (Fault-Hypothesis Short-Circuit ?component)

    (Schema ?component
         (instance-of component:power-sink
         (source ?source& none)
    (Source-sensor ?sensor))

    (Schema ?sensor
         (instance-of sensor)
         (load-sensor yes)
         (Faulted-Voltage ?value))

    (Test      (> ?value 10))

    (Assert (Inconsistent-hypothesis-v Short-circuit ?component)))
```

"If a component is a power sink and its voltage is greater than 10 then discard a hypothesis that there is a short circuit for the component."

```
(Defrule Inconsistent-Load-Resistive-Short-Circuit
    (declare (salience 610))
    (Fault-Hypothesis Resistive-Short-Circuit  ?component)

    (Schema ?component
         (instance-of component)
         (source ?source& none)
         (Source-sensor ?sensor))

    (Schema ?sensor
         (instance-of sensor)
         (load-sensor yes)
         9Faulted-Voltage ?value))

    (Test      (< ?value 10))

    (Assert (Inconsistent hypothesis v Resistive Short Circuit ?component)))
```

"If a component is a load (power sink) and its source sensor registers a voltage of less than 10 then discard any hypothesis of a short circuit at that component.

```
(Defrule INconsistent-Load-Closed Relay
    (declare (salience 610))
    (Fault-Hypothesis Closed Relay  ?component)

    (Schema ?sensor
         (instance-of sensor)
         (load sensor yes)
         (Potential-Relay Sink ?component)
         (Faulted-Voltage ?value))

    (Test      (< ?value 20))

    (Assert (inconsistent hypothesis-v Closed Relay ?component)))
```

"If the sink for a sensor is a relay and the voltage at the sensor is less than 20 then discard any hypothesis that there is a closed relay fault at the relay."

```
(Defrule Init Source Prop Open
    (declare (salience 568))
    (Fault Hypothesis Open Circuit:Open Relay ?component)

    (Schema ?component
         (instance of component)
         (source ?source & none))

    (Assert (Propagate to sc ?component ?source (Current Zero))))
```

"Voltage is zero across properly opened relays between a component and its source."

```
(Defrule Init-Sink-Prop-Open
    (declare (salience 568))
    (Fault-Hypothesis Open-Circuit:Open-Relay ?component)

    (Schema ?component
         (instance-of component)
         (sink ?sink& ~none))

    (Assert (Propagate-to-sk ?component ?sink (Current Zero))))

"Same as above except for relays between a component and its sink."
(Defrule Init-Source-Prop-Open-From-PSink
    (declare (salience 558))
    (Fault Hypothesis Open Circuit ?component)

    (Schema ?component
         (instance of power-sink)
         (Sink none)
         (source ?source& ~none))

    (Assert (Propagate-to-sc ?component ?source (Current Zero))))

"If a power sink has a source and it has been hypothesized that there is an open
    ciruit for the power sink then propogate that hypothesis to the source."
```

```
(Defrule Init-Sink-Prop-Open-From-PSource
    (declare (salience 558))
    (Fault-Hypothesis Open-Circuit ?component)

    (Schema ?component
         (instance-of power-source)
         (Source none)
         (sink ?sink&~none))

    (Assert (propagate-to-sk ?component ?sink (Current Zero))))
```

"If a power source has a sink and it has been asserted that there is an open circuit for the power source then propagate the hypothesis to the sink for the power source."

```
(Defrule Init-Source-Prop-Short
    (declare (salience 548))
    (Fault-Hypothesis Short-Circuit ?component)
    (not (Inconsistent-hypothesis-v Short-circuit ?component))

    (Schema ?component
         (instance-of component)
         (source ?source&~none))


    (Assert (Porpagate-to-sc ?component ?source (Current High))))
```

"If a short circuit is hypothesized for a load, then propagate back the high current reading."

```
(Defrule Init-Sink-Prop-Short
    (declare (salience 548))
    (Fault-Hypothesis Short-Circuit ?component)
    (not (Inconsistent-hypothesis-v Short-circuit ?component))

    (Schema ?component
         (instance of component)
         (sink ?sink&`none))

    (Assert (Propagate-to sk ?component ?sink (Current Zero))))
```

"If a component has been hypothesized as having a short ciruit and the component has a sink then propagate the hypothesis to the sink."

```
(Defrule Init-Sink-Prop-Short-From-PSource
    (declare (salience 548))
    (Fault-Hypothesis Short-Circuit ?component)

    (Schema ?comopnent
         (instance of power-source)
         (source none)
         (sink ?sink&`none))

    (Assert (Propagate to sk ?component ?sink (Current Zero))))
```

"If a short circuit has been hypothesized for a power source, then propagate forward the zero current to the sink for the power source."

```
(Defrule Init-Source-Prop-Short-From-PSink
    (declare (salience 538))
    (Fault-Hypothesis Short-Circuit ?component)
    (not (inconsistent-hypothesis-v Short-circuit ?component))

    (Schema ?component
         (instance-of power-sink
         (sink none)
         (Source ?source&~none))

    (Assert (propagate-to-sc ?component ?source (Current High))))

If a short circuit has been hypothesized for a power sink, then propagate
    back the high current from the sink to its source."
(Defrule Init-Source-Prop-RShort
    (declare (salience 528))
    (Fault-Hypothesis Resistive Short-Circuit ?component)
    (not (Inconsistent-hypothesis-v Resistive-Short-Circuit ?component))

    (Schema ?component
         (instance-of component)
         (source ?source&~none))

    (Assert (Porpagate-to-sc ?component ?source (Current High))))

"If a resisitive short circuit has been hypothesized for a component, then
    propagate back the high current to the source for the component."
```

```
(Defrule Init-Sink-Prop-RShort
    (declare (salience 528))
    (Fault-Hypothesis Resistive-Short-Circuit ?component)
    (not (Inconsistent-hypothesis-v Resistive-Short-circuit ?component))

    (Schema ?component
         (instance-of component)
         (Sink ?sink&~none))

    (Assert (Porpagate-to-sk ?component ?sink (Current Low))))
```

"If a resisitive short circuit has been hypothesized for a component, then propagate forward the low current reading to its sink."

```
(Defrule Init-Sink-Prop-RShort-From-Psource
    (declare (salience 518))
    (Fault-Hypothesis Resistive-Short-Circuit ?component)

    (Schema ?component
         (instance-of power-source0
         (source none)
         (sink ?sink&~none))

    (Assert (Propagate to sk ?component ?sink (Current Zero))))
```

"If a resisitive short circuit has been hypothesized for a power source, then propagat forward to the sink for the source the zero current."

```
(Defrule Init-Source-Prop-RShort-From-Psink
    (declare (salience 518))
    (Fault-Hypothesis Reso=istive-Short-Circuit ?component)
    (not (Inconsistent-Hypothesis-v Resistive-Short-circuit ?component))

    (Schema ?component
         (instance-of power-sink)
         (sink none)
         (source ?source&~none))

    (Assert (Propagate-to-sc ?component ?source (Current High))))
```

"If a resistive short circuit has been hypothesized for a power sink, then propagate back the high current reading to the source for the sink."

ORIGINAL PAGE IS
OF POOR QUALITY

```
(Defrule Continue-Propagation-Sc
    (declare (salience 600))
    (propagate-to-sc ?from ?component (Current ?effect))
    (Schema ?component
         (instance-of component)
         (Source ?source&~none)
         (multiple-Sources ?)
         (Multiple Sinks no))

    (not (Schema ?component
         (instance-of sensor)))

    (Fault-Hypothesis ?fault ?location)
    (not (Inconsistent-hypothesis ?fault ?location))

"See next rule."


(Defrule Continue-Propagation-sk
    (declare (salience 599))
         (Propagate-to-sk ?from ?component (Current ?effect))
         (Schema ?component
              (instance-of component)
              (Sink ?sink&~none)
              (Multiple Sources no)
              (Multiple-Sinks ?))
    (not (Schema ?comonent
              (instance of sensor)))

    (Fault Hypothesis ?fault ?location)
    (not (Inconsistent hypothesis ?fualt ?location))


"These two rules continue the propagation process initiated by the preceding
    rules."
```

ORIGINAL PAGE IS
OF POOR QUALITY


```
(Defrule Continue-Propagation-sk-2-sources
    (declare (salience 599))
         (Propagate-to-sk ?from ?component (Current zero))
         (Schema ?component
              (instance-of component)
              (Sink ?sink&~none)
              (Multiple-Sources yes)
              (Multiple-Sinks no)
              (I-Eq ((+ S? ?other-sources&~?from $?) (+ ?sink))))

    (not (Schema ?component
              (instance-of sensor)))

    (Fault-Hypothesis ?fault ?location)
    (not (Inconsistent-hypothesis ?fault ?location))

    (Assert (Propagate-to-sc ?comopnent ?other-source (Current high)))
    (Assert (Propagate-to-sk ?component ?sink (Current Low))))


(Defrule Continue-Propagation-source-2-sks-high
    (declare (salience 600))
    (Propagate-to-sc ?from ?component (Current high))
    (not (Path-Already Followed Propagate-to-sc ?component high))
    (Schema ?component
         (instance-of component)
         (Source ?source&~none)
         (Multiple-Sources no)
         (Multiple-Sinks yes)
         (I-Eq ((+ ?source) (+ $? ?other-sink& ?from $?))))

    (not (Schema ?component
         (instance-of sensor)))
    (Fault-Hypothesis ?fault ?location)

    (not (Inconsistent-hypothesis ?fault ?location))
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
(Defrule Continue-Propagation-source-2-sks-zero
    (declare (salience 600))
    (Porpagate-to-sc ?from ?component (Current zero))
    (Schema ?component
         (instance-of component)
         (Source ?source&~none)
         (Multiple-Sources no)
         (Multiple-Sinks yes)
         (I-Eq ((+ ?source) (+ $? ?other-sink&~?from $?))))

    (not (Schema ?component
         (instance-of sensor)))

    (Fault-Hypothesis ?fault ?location)

    (not (Inconsistent-hypothesis ?fault ?location))


(Defrule Continue-Propagation-source-2-sks-low
    (declare (salience 600))
    (Propagate-to-sc ?from ?component (Current low))
    (not (Path-Already Followed Propagate-to-sc ?component low))
    (Schema ?component
         (instance-of component)
         (Source ?source&~none)
         (Multiple-Sources no)
         (Multiple-Sinks yes)
         (I-Eq ((+ ?source) (+ $? ?other-sink&~?from $?))))

    (not (Schema ?component
         (instance-of sensor)))

    (Fault-Hypothesis ?fault ?location)

    (not (Inconsistent-hypothesis ?fault ?location))
```

ORIGINAL PAGE IS
OF POOR QUALITY


```
(Defrule Sensor-Consistency-Check-Sc-consistent
    (declare (salience 600))
    (Propagate-to-Sc ?from ?component (Current ?pred-effect))
    (Schema ?component
         (instance-of sensor)
         (Qual-Current ?qual-current
         (Source ?source&~none)
         (Multiple-Sources no)
         (Multiple-Sinks no))

    (Consistent-Effect ?pred-effect ?qual-current )
    (Fault-Hypothesis ?fault ?location)
    (Not (Inconsistent-hypothesis ?fault ?location))


(Defrule Sensor-Consistency-Check-Sc-fadeout
    (declare (salience 600))
    (Propagate-to-Sc ?from ?component (Current ?pred-effect))
    (Schema ?component
         (instanct-of sensor)
         (Qual-Current ?qual-current)
         (Source ?source&~none)
         (Multiple-Sources no)
         (Multiple-Sinks no))

    (not (Consistent-Effect ?pred-effect ?qual-current ))
    (Fadeout-Effect-Sc-Prop ?pred-effect ?qual-current)

    (Fault-Hypothesis ?fault ?location)
    (not (Inconsistent-hypothesisi ?fault ?location))
```

```
(Defrule Sensor-Consistency-Check-Sc-noise
    (declare (salience 600))
    (Porpagate-to-Sc ?from ?component (Current ?pred-effect))
    (Schéma ?component
          (instance-of sensor)
          (Qual-Current ?qual-current)
          (Source ?source&~none)
          (Multiple-Sources no)
          (Multiple-Sinks no))
    (not (Consistent-Effect ?pred-effect ?qual-current ))
    (not (Fadeout-Effect-Sc-Prop ?pred effect ?qual-current))

          (Noise-Effect ?pred-effect ?qual-current)
    (Fault-Hypothesis ?fault ?location)
    (not (Inconsistent-hypothesis ?fault ?location))


(Defrule Sensor-Consistency-Check-sk consistent
    (declare (salience 599))
    (Propagate-to-sk ?from ?component (Current ?pred-effect))
    (Schema ?component
          (instance-of sensor)
          (Qual-Current ?qual-current)
          (Sink ?sink&~none))
    (Consistent-Effect ?pred-effect ?qual-current )

    (Fault-Hypothesis ?fault ?location)
    (not (Inconsistent-hypothesis ?fault ?location))
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
(Defrule Sensor-Consistency-Check-sk-fadeout
    (declare (salience 599))
    (Propagate to sk ?from ?component (Current ?pred effect))
    (not (Path-Already-Followed Propagate to sk ?component ?pred-effect))
    (Schema ?component
         (instance-of sensor)
         (Qual Current ?qual-current)
         (Sink ?sink&~none))
    (Not (Consistent Effect ?pred effect ?qual current ))

         (Fadeout-Effect-Sk-Prop ?pred-effect ?qual-current)

    (Fault Hypothesisi ?fault ?location)
    (not (Inconsistent-hypothesis ?fault ?location))

(Defrule Sensor Consistency Check sk noise
    (declare (salience 599))
    (Propagate to sk ?from ?component (Current ?pred effect))
    (Schema ?component
         (instance of sensor)
         (Qual Current ?qual-current)
         (Sink?sink&~none))
    (not (Consistent-Effect ?pred-effect ?qual current ))
    (not (Fadeout Effect Sk Prop ?pred effect ?qual current))
         (Noise Effect ?pred-effect ?qual-current)
    (Fault Hypothesis ?fault ?location)
    (not (Inconsistent-hypothesis ?fault ?location))
```

```
(Defrule Sensor-Inconsistency-Check-source
    (declare (salience 601))
    (Propagate-to-Sc ?from ?component (Current ?pred effect))
    (Schema ?component
         (instance-of sensor)
         (Qual-Current ?qual-current)
         (Sink ?sink&~none))
    (Not (Consistent-Effect ?pred effect ?qual-current ))
    (Not (Fadeout-Effect-Sk-Prop ?pred effect ?qual current))

    (Not (Noise-Effect ?pred-effect ?qual-current ))
    (Fault Hypothesis ?fault ?location)
    (not (Inconsistent-hypothesis ?fault ?location))


(Defrule Sensor-Inconsistency-Check-sk
    (declare (salience 601))
    (Propagate-to-sk ?from ?component (Current ?pred effect))
    (Schema ?component
         (instance-of sensor)
         (Qual-Current ?qual-current)
         (Sink ?sink&~none))

    (Not (Consistent-Effect ?pred effect ?qual-current ))
    (Not (Fadeout-Effect-Sk-Prop ?pred effect ?qual-current))

    (Not (Noise-Effect ?pred-effect ?qual-current ))
    (Fault Hypothesis ?fault ?location)
    (not (Inconsistent-hypothesis ?fault ?location))
```

APPENDIX E

Hardware Description and Drawings

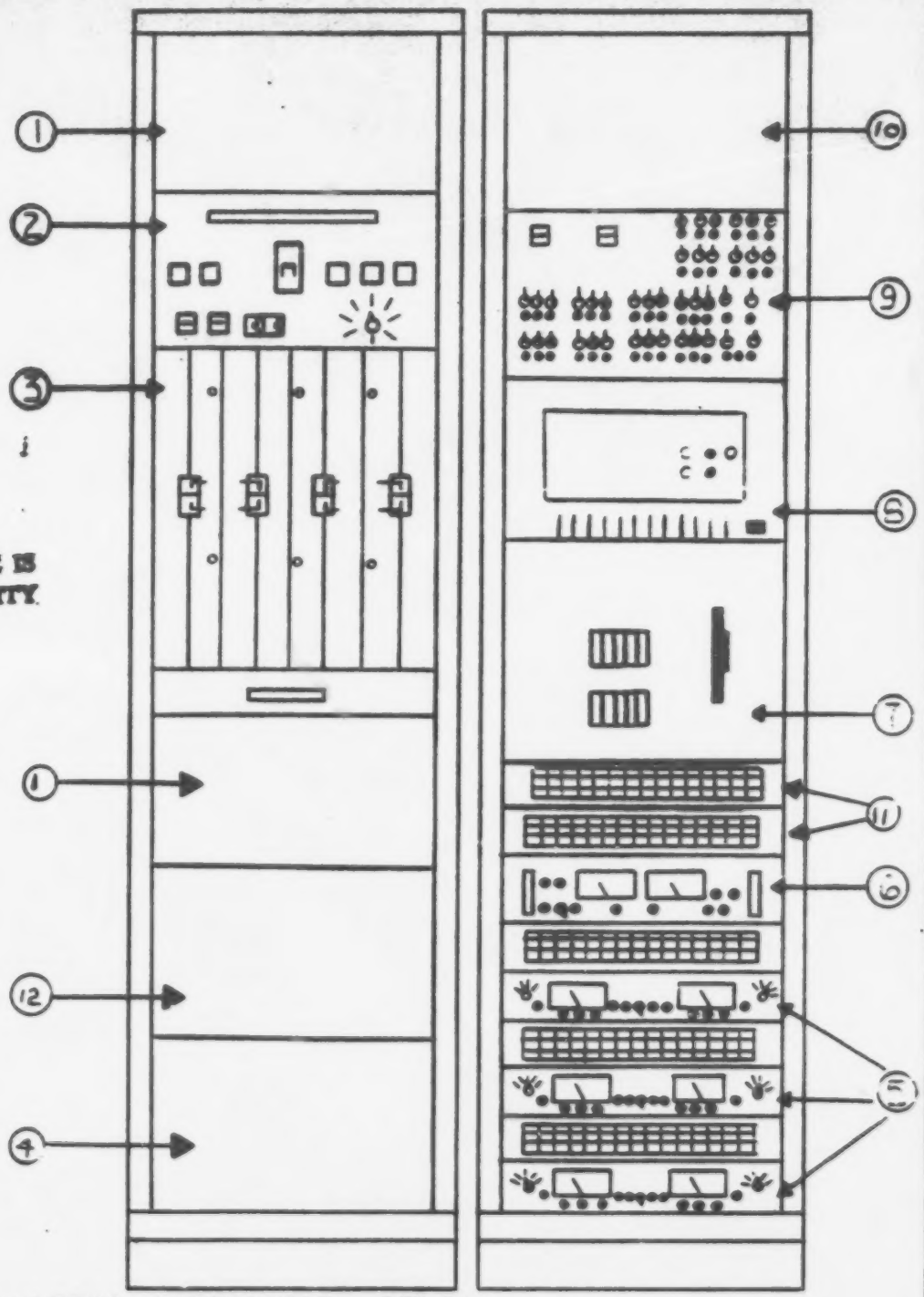
SUBSYSTEM AUTOMATION STUDY BREADBOARD

Drawing #PL849PABB1000

Item #	Part#	Description	MFG	QTY
1	P-10	10.5" Panels	Optima	3
2	849PABB1500	Main Control Assembly		1
3	849PABB1600	Power System Assembly		1
4	849PABB1700	Battery System Assembly		1
5	HP6255A-010	Power Supply	Hewlett Packard	3
6	HP6264B-010	Power Supply	Hewlett Packard	1
7	849PABB1900	Microcomputer Disk Drive	Intel	1
8	849PABB1400	Microcomputer Assembly	Intel	1
9	849PABB1300	Fault Insertion Assembly		1
10	849PABB1200	Map Assembly		1
11	G-319	Ventillation Grille	Optima	5
12	849PABB1100	Voltage Regulator		1

Numbers which end in 1 are schematics. Drawings beginning in PL are parts lists for the assembly drawings which follows:

1000 - Top Level assembly
11 - First assembly beneath top level
12 - Second assembly beneath top level
1110 - First assembly beneath second level

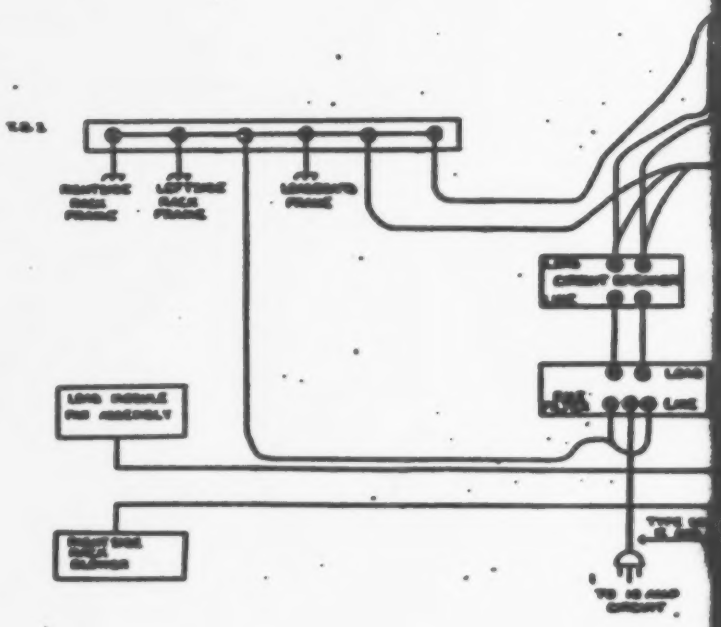


ORIGINAL PAGE IS
POOR QUALITY

POWER SUBSYSTEM AUTOMATION STUDY
SYSTEM ASSEMBLY DRAWING

	SIZE	FSCM NO.	DWG NO.	REV
E-2	A		849PABB1000	
	SCALE 1-1	PAGE	SHEET 2F1	

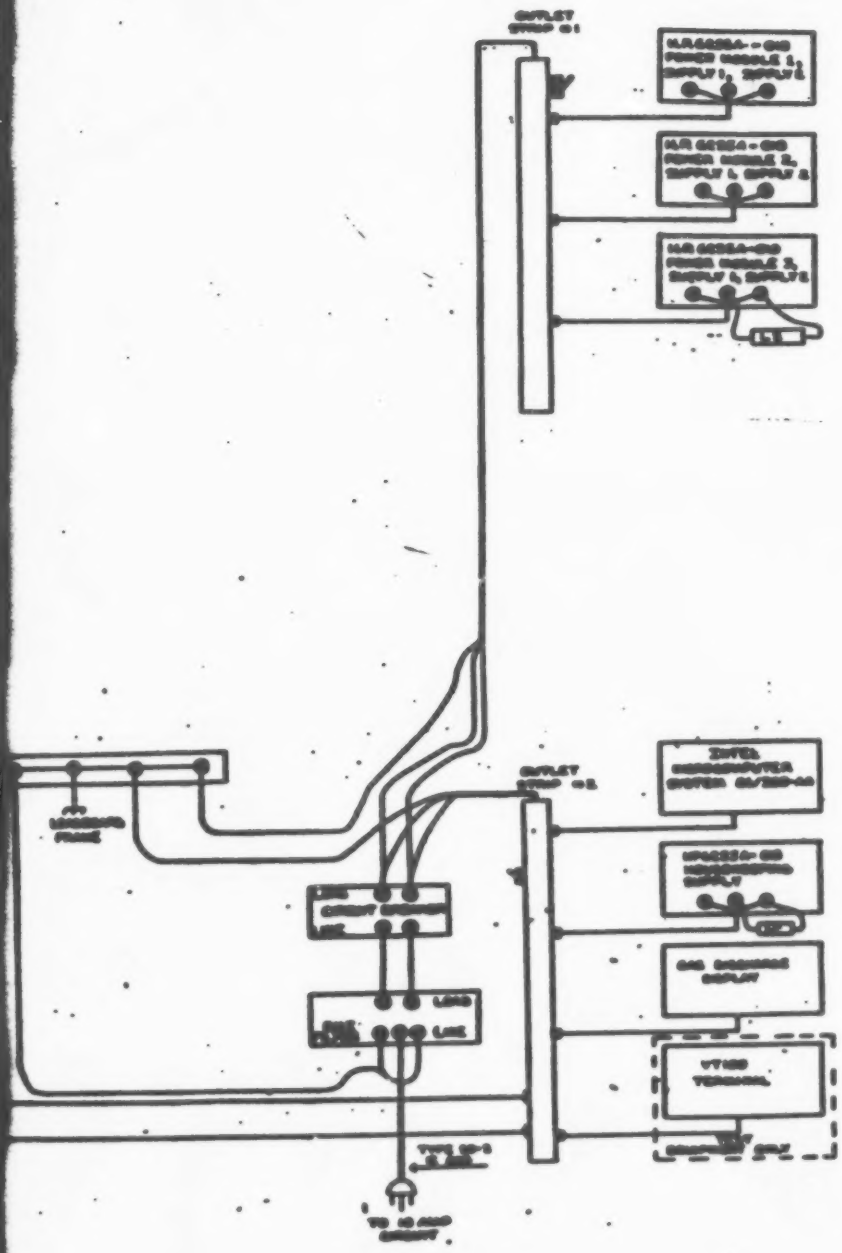
ORIGINAL PAG
OF POOR QUA



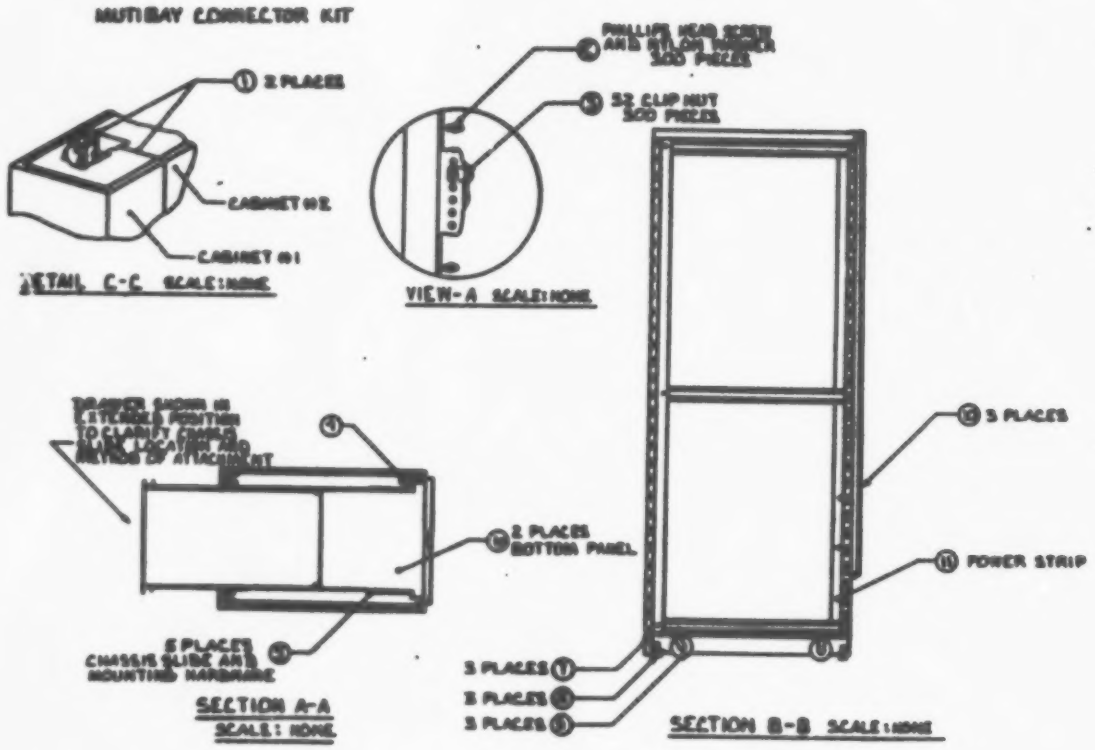
FOLDOUT FRAME

E-3

ORIGINAL PAGE IS
OF POOR QUALITY



FOLDOUT FRAME



ORIGINAL PAGE IS OF POOR QUALITY

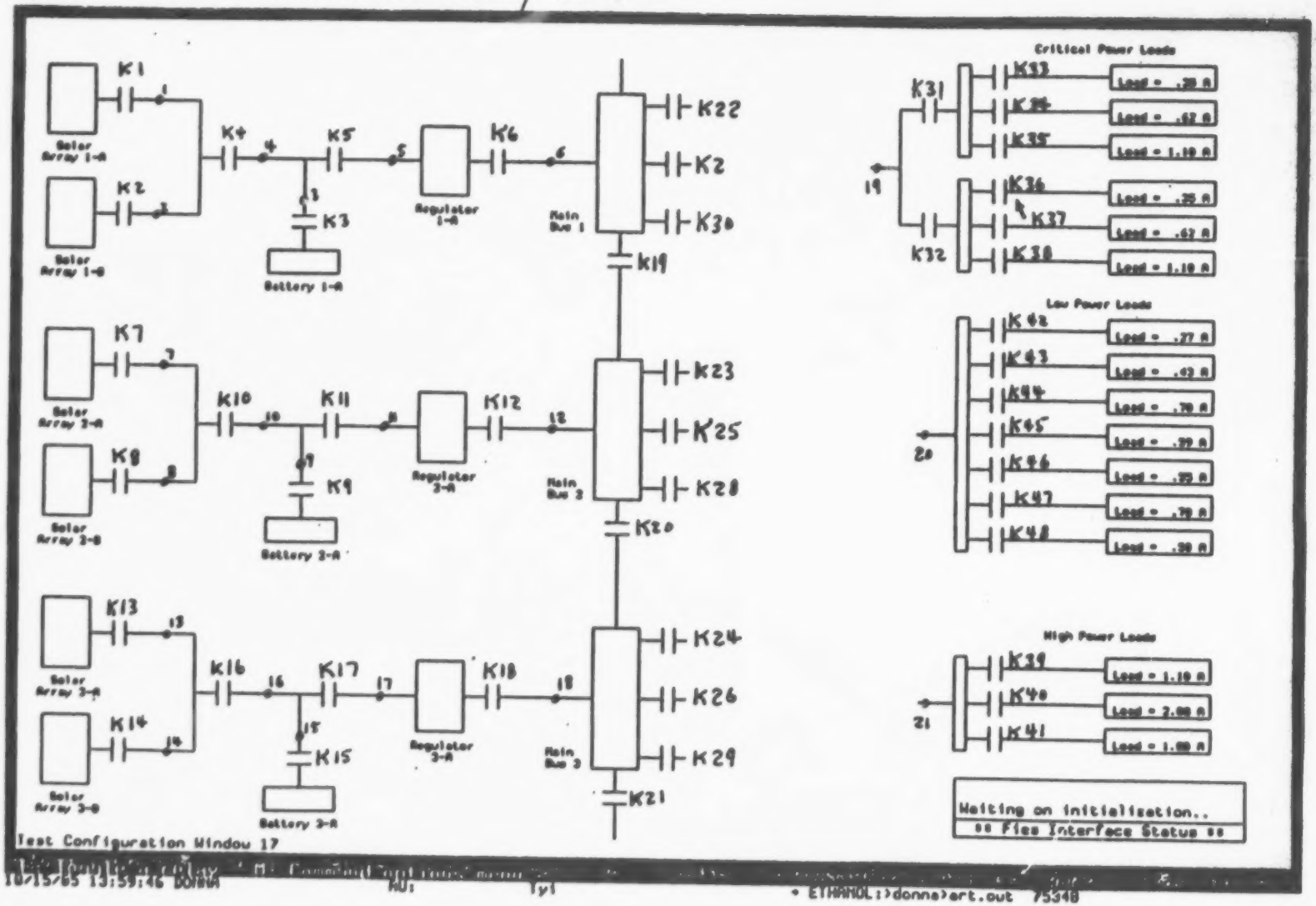
POWER SUBSYSTEM AUTOMATION STUDY-
RACK ASSEMBLY

C

NONE

E-4

RELAY AND SENSOR POSITIONS/NUMBERS



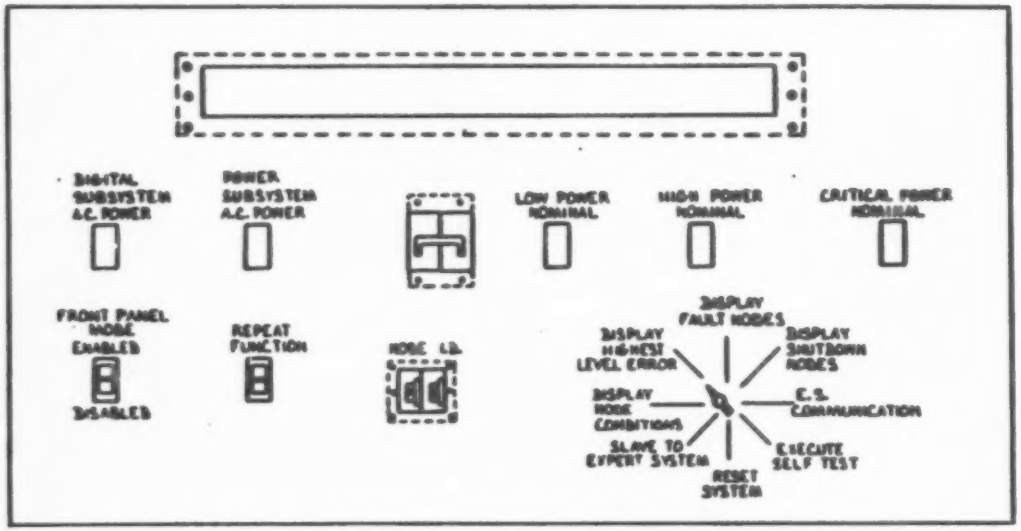
ORIGINAL PAGE IS
OF POOR QUALITY

MAIN CONTROL ASSEMBLY

Drawing #PL849PABB1500

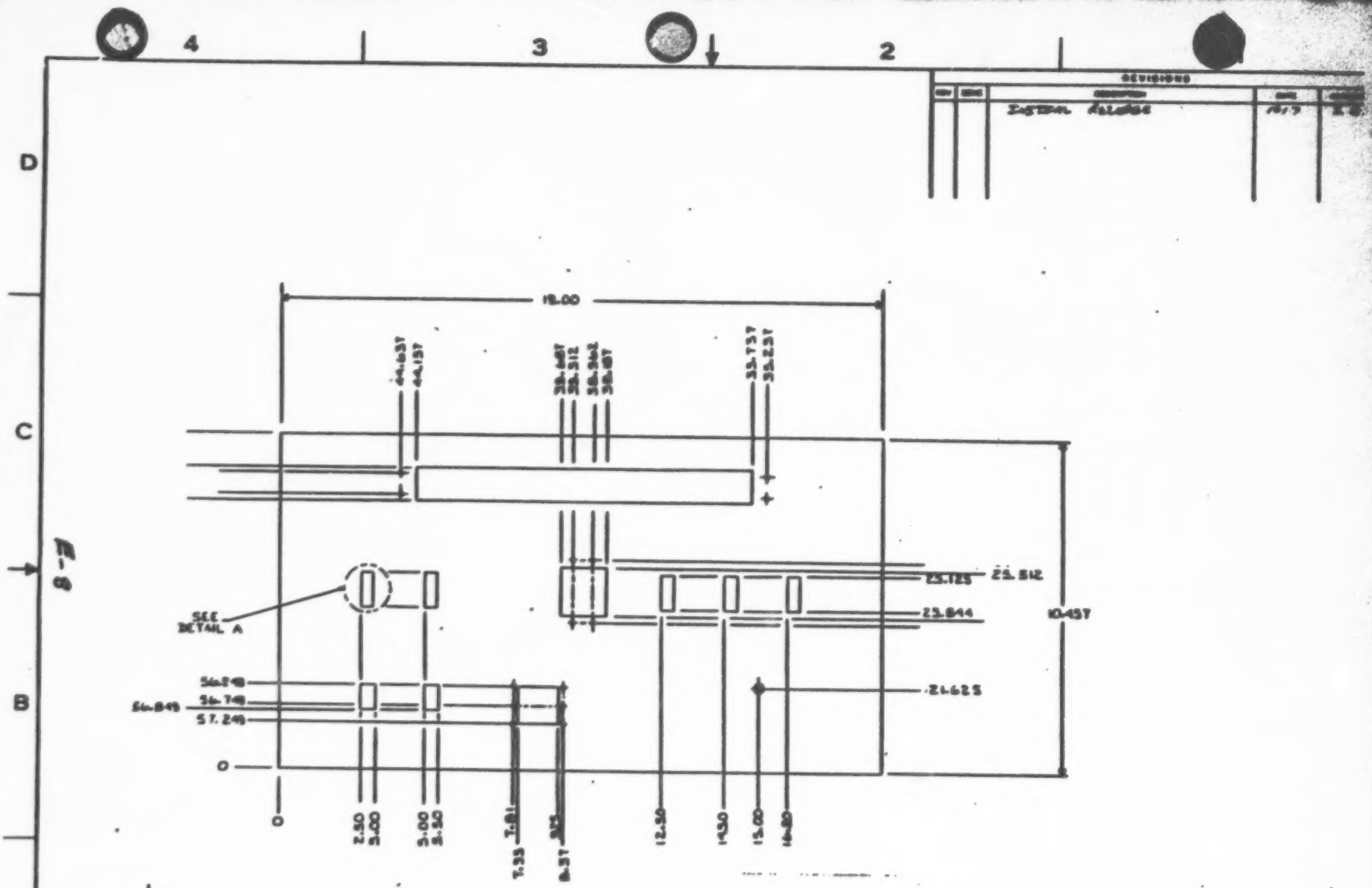
Item #	Part #	Description	MFG	QTY
1	P-10	10.5" X 19" Panel	Optima	1
2	W424-1050	Alphanumeric Display	Cherry Electric	1
3	AM2-A3-A-10-2	Circuit Breaker	Heinemann	1
4	APF2021S	Line Filter	Cornell-Dubilies	1
5	31-2111T	AC Indicator Lamps	Leecraft	2
6	31G3-2111T	DC Indicator Lamps	Leecraft	3
7	572-1321-0804-040	Momentary Paddle Switch	Dialight	1
8	572-1121-0804-040	2 Position Paddle Switch	Dialight	1
9	PAZ001	Rotary Switch	Centralab	1
10	3-2-11-1-4-3-0-1	Thumbwheel Switch	C & K	1
11	RH019,150	1% Current Limiting Resistors	Dale	3
12	RC07GF201J	200,1/4 Volt Resistor 5%	Allen-Bradley	3
13	Th-112-48L	Lettering Set	Tranfertech	3

REVISIONS				
REV	ZONE	DESCRIPTION	DATE	
		DIGITAL RELEASE	10/7	E,



POWER SUBSYSTEM AUTOMATION STUDY
MAIN CONTROL PANEL -
ASSEMBLY DRAWING AND DRILL DETAIL

MARTIN MARIETTA CORPORATION	SIZE C	FSCM NO 84236	DWG NO 849PABB.500
DRAWN BY Jill Butler			
18 Feb 85	SCALE NONE		SHEET 1 OF 2



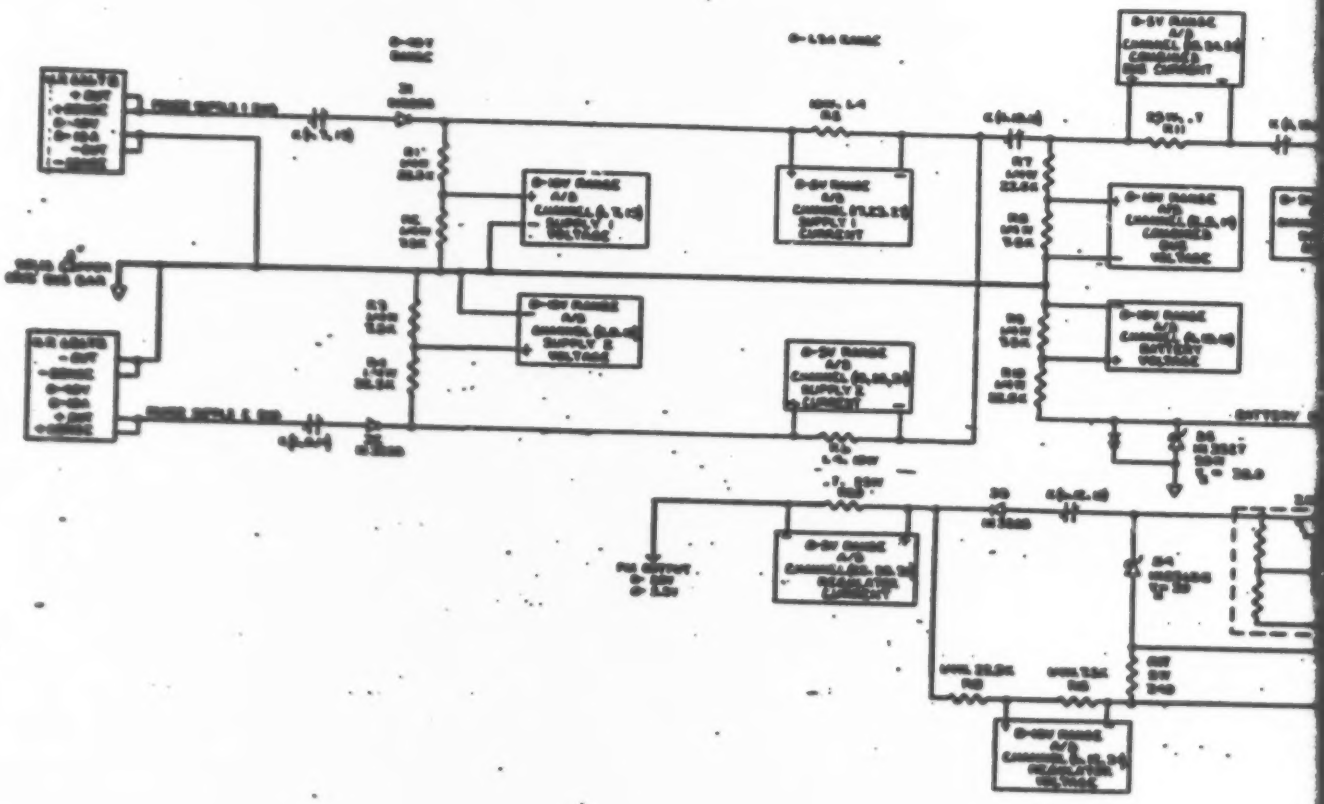
POWER SUBSYSTEM AUTOMATION STUDY
MAIN CONTROL PANEL-
ASSEMBLY DRAWING AND DRILL DETAIL

MARTIN MARIETTA CORPORATION	C	84236	849PABB1500
22FEB85	SCALE NONE		SHEET 2 OF 2

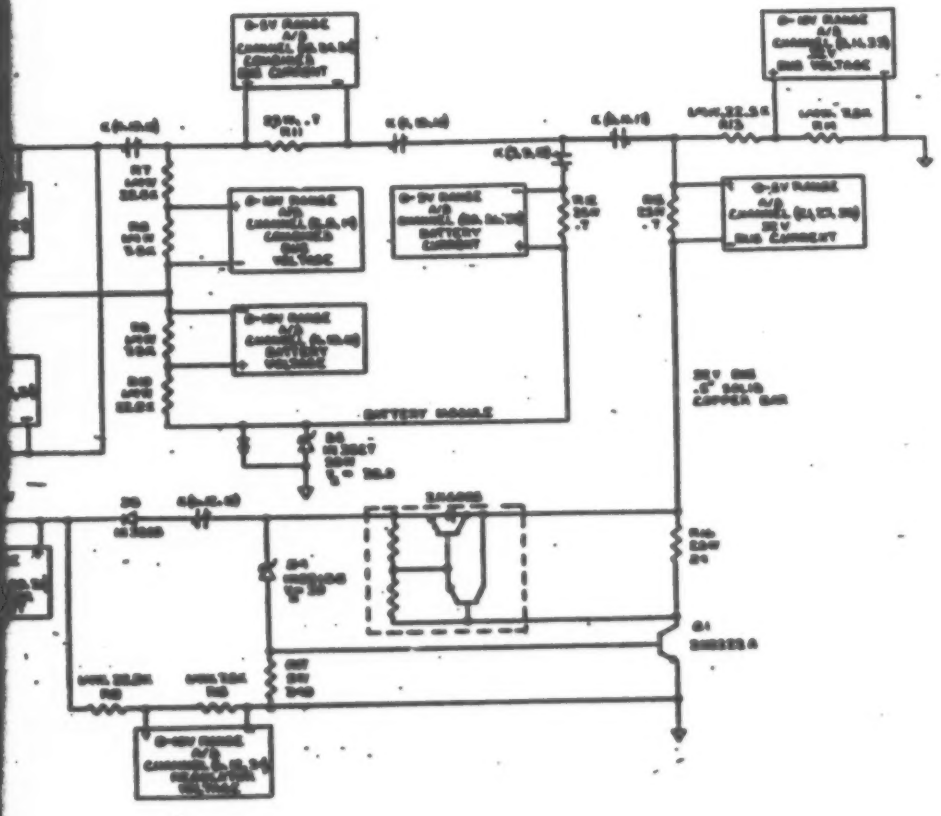
E-8

FOLDOUT FRAME

ORIGINAL PAGE IS
OF POOR QUALITY

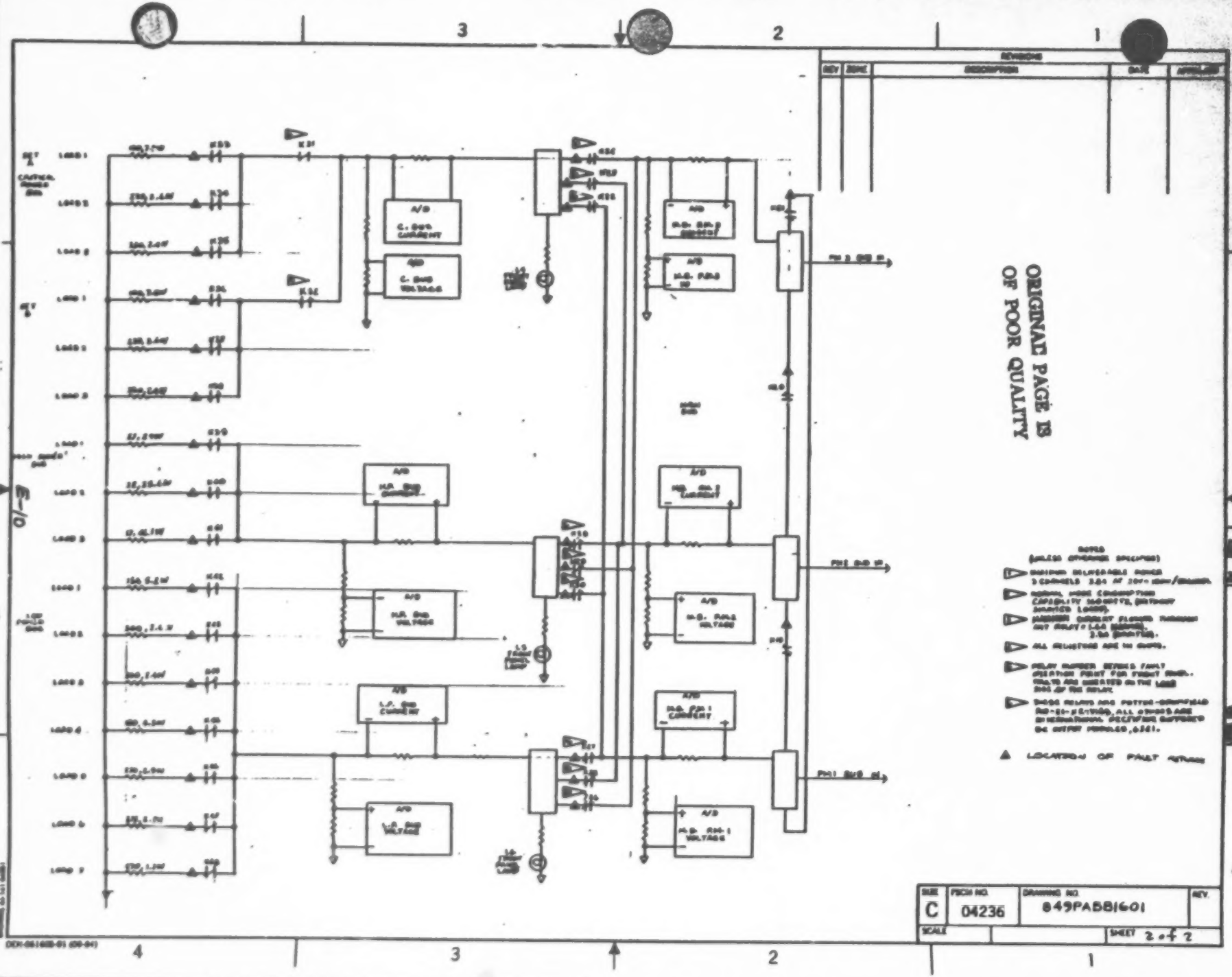


FOLDOUT FRAME



POWER MODULE 1 OF 3
BLOCK DIAGRAM

Sheet 1 of 2



ORIGINAL PAGE IS OF POOR QUALITY

POWER SYSTEM - BOARD 1

Drawing #PL849PABB1610

Item #	Part #	Description	MFG	QTY
1		T4 Aluminum	MMC	1
2		2/56 Screw	NL Fasteners	4
3	Module 39			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%,50W Resistor	Dale	1
	NH-50,27 ohm	1%,50W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt,27ohm, 1% Resistor	Dale	1
	RC07GF271J	1/4 Watt,270ohm,5% Resistor	Allen-Bradley	1
		Capton Tape		1
4	Module 48			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	NH-5,330 ohm	1%, 5W Resistor	Dale	3
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt,270ohm, 5%	Allen-Bradley	1
		Capton Tape		1
5	Module 47			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-50,50 ohm	1%, 5W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1

POWER SYSTEM - BOARD 1
Con't

Drawing #PL849PABB1610

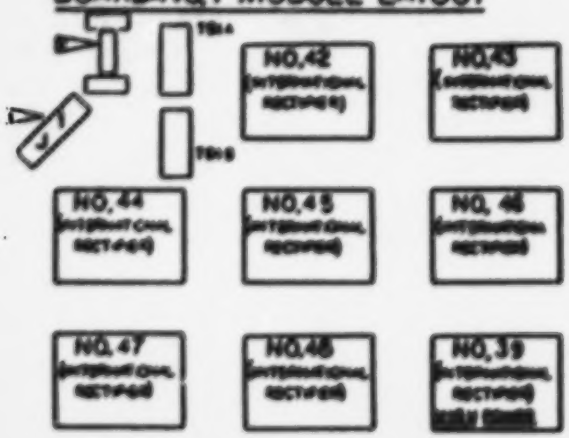
Item #	Part #	Description	MFG	QTY
6	Module 46			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	NH-50,27 ohm	1%, 50W Resistor	Dale	1
	RH-5,40 ohms	1%, 5W, Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1
7	Module 45			
	6321	Solid State Relay	Cryden	1
	RH-5,180ohms	1%, 5W Resistor	Dale	3
	NH-50,14ohms	1%, 50W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1
8	Module 44			
	6321	Solid State Relay	Cryden	1
	NH-50,14ohms	1%, 50W Resistor	Dale	1
	RH-50,50ohms	1%, 50W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1

POWER SYSTEM - BOARD 1
Con't

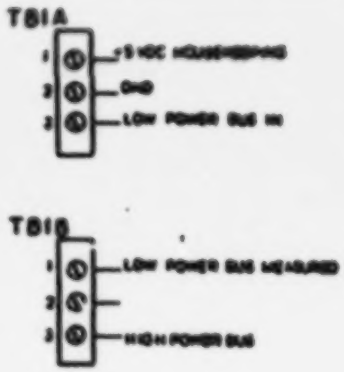
Drawing #PL849PABB1610

Item #	Part #	Description	MFG	QTY
9	Module 43			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5,270ohm	1%, 50W Resistor	Dale	3
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1
10	Module 42			
	6321	Solid State Relay		1
	NH-50,14 ohm	1%, 50W Resistor		1
	RH-10,150ohm	1%, 10W Resistor		1
	TVA-1305.5	20VF, 50V Capacitor	Sprague	1
	2N6396	SCR		1
	2N6039	Power Transistor		1
	2N2222A	Small Signal Transistor		1
	RS2027	1 Watt, 27 ohm, 1%		1
	RC07GF271J	1/4 Watt, 270 ohm, 5%		1
		Capton Tape		1
11	Measurement Network			
	RH-5,1ohm	1%,5W Resistor	Dale	2
	RNC6507501F	7.5Kohm, 1% Resistor	Dale	2
	RNC65N2262FS	22.6Kohm, 1% Resistor	Dale	2
	300TE1305	20nF Capacitor	Sprague	2
12	72103	Terminal Blocks	Vernitron	2
13	DB25S	'D' Type, 25 Pin Female Connector	Amphenol	1

BOARD NO.1 MODULE LAYOUT



BOARD NO.1 TERMINAL BLOCK DESCRIPTION



ORIGINAL PAGE IS OF POOR QUALITY

NOTES:

- CONNECTOR J1 IS A 25-PIN D'TYPE SIGNAL CONNECTOR
- MEASUREMENT NETWORK

BOARD NO.1 J1 PIN DEFINITION

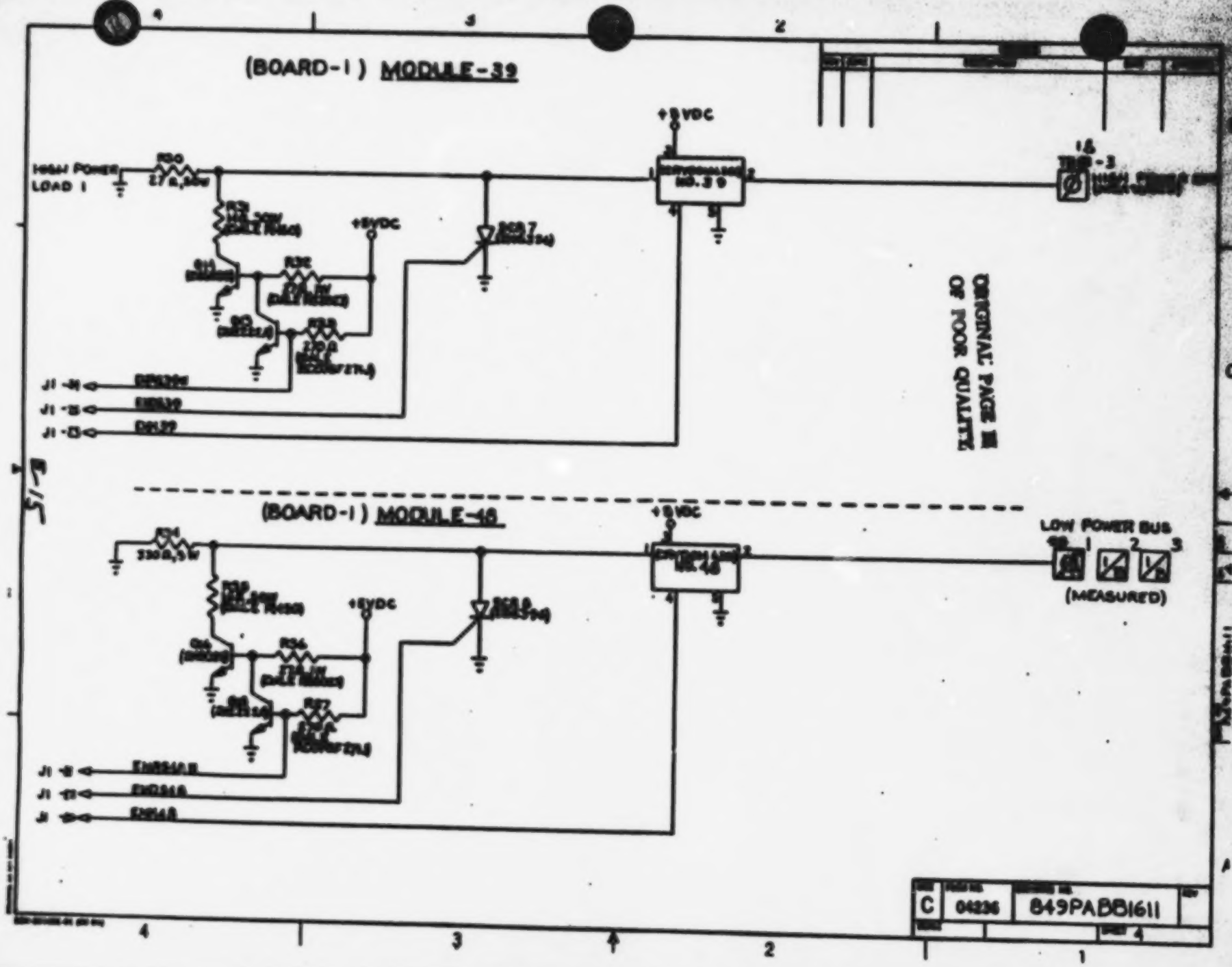
PIN NO.	FOR RELAY NO.	SIGNAL
1	42	ENR42
2	42	ENR842
3	42	END842
4	43	ENR43
5	43	ENR843
6	43	END843
7	44	ENR44
8	44	ENR844
9	44	END844
10	45	ENR45
11	45	ENR845
12	45	END845
13	46	ENR46
14	46	ENR846
15	46	END846
16	47	ENR47
17	47	ENR847
18	47	END847
19	48	ENR48
20	48	ENR848
21	48	END848
22	39	ENR39
23	39	ENR839
24	39	END839
25	39	END839

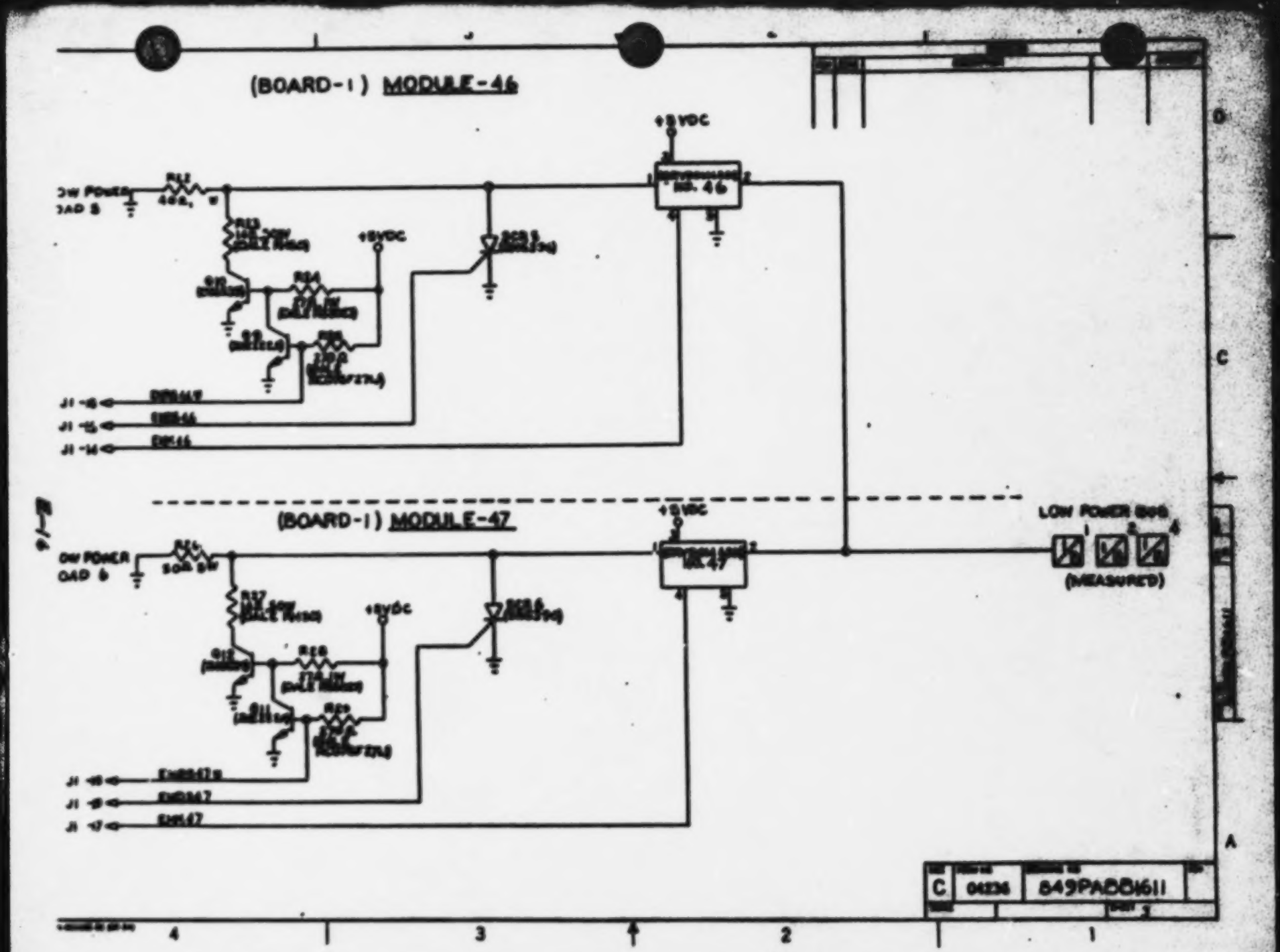
MARTIN MARIETTA CORPORATION

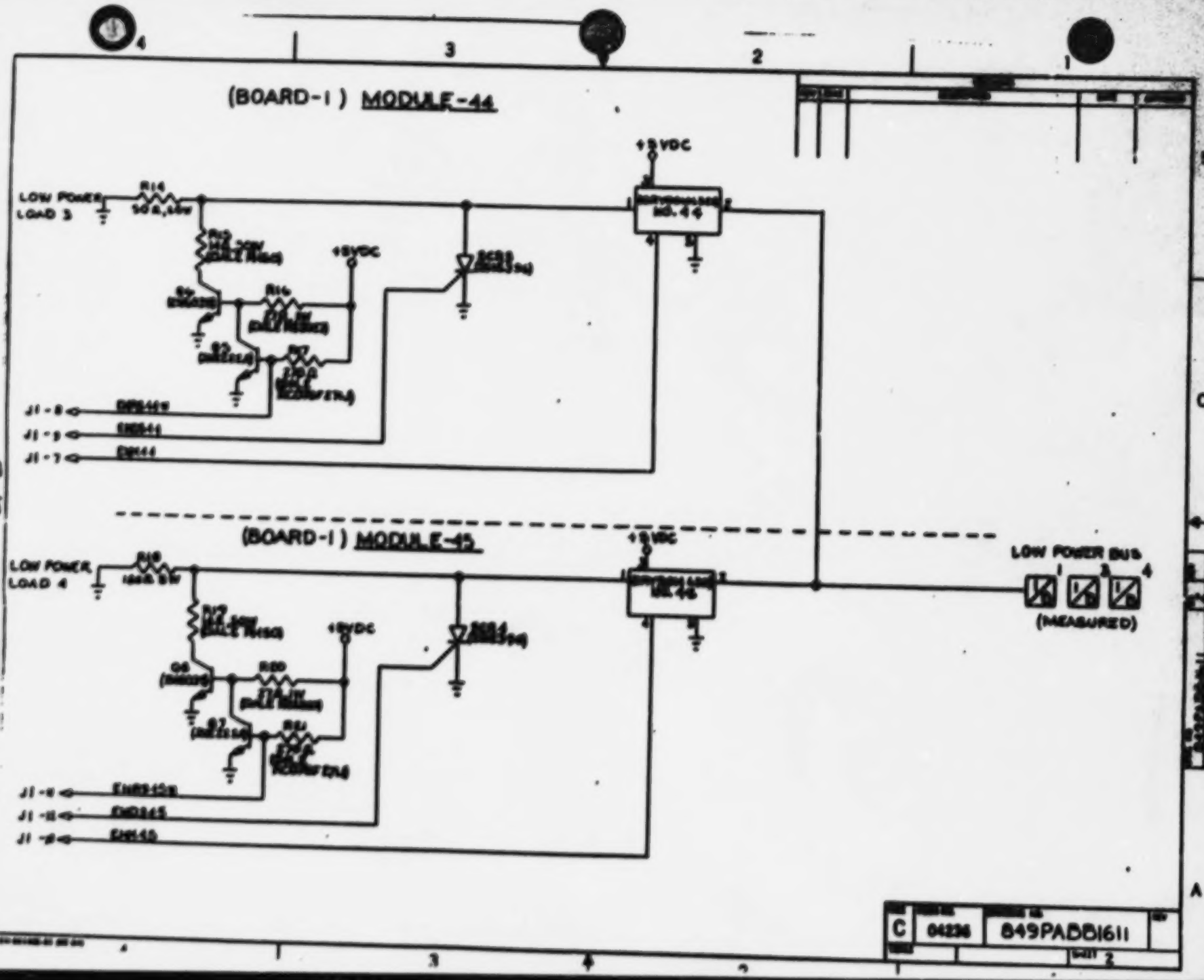
BOARD NO.1 LAYOUT AND SCHEMATIC, FIES PROGRAM

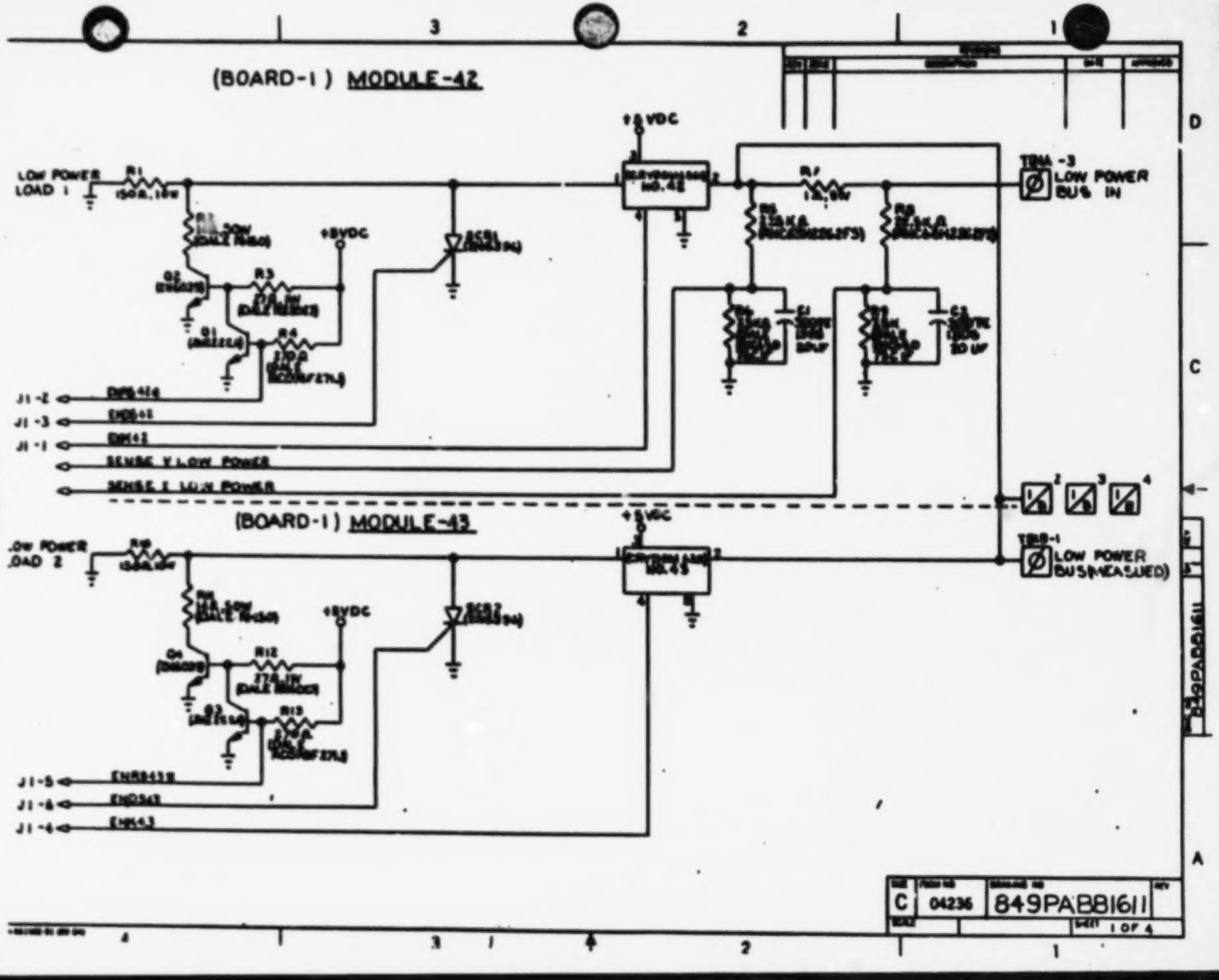
C 849PABE16IC

203605


(BOARD-1) MODULE-39
(BOARD-1) MODULE-46
+5VDC
HIGH POWER LOAD 1
LOW POWER BUS
(MEASURED)
ORIGINAL PAGE IS OF POOR QUALITY
C
04236
849PABB1611
PAGE 4


(BOARD-1) MODULE-46
+5VDC
NO. 46
(BOARD-1) MODULE-47
+5VDC
NO. 47
LOW POWER BUS
(MEASURED)
C
04236
849PABB1611


(BOARD-1) MODULE-44
LOW POWER LOAD 3
R14
+5VDC
NO. 44
J1-8
J1-9
J1-7
(BOARD-1) MODULE-45
LOW POWER LOAD 4
R18
+5VDC
NO. 45
LOW POWER BUS
(MEASURED)
J1-11
J1-12
J1-13
C
04236
849PABB1611
SHEET 2


(BOARD-1) MODULE-42
LOW POWER LOAD 1
R1
+5VDC
R3
R4
Q1
Q2
SCR1
J1-2
J1-3
J1-1
SENSE Y LOW POWER
SENSE X LOW POWER
TB1A-3
LOW POWER BUS IN
(BOARD-1) MODULE-43
LOW POWER LOAD 2
R10
R12
R13
Q3
Q4
SCR2
J1-5
J1-6
J1-4
TB1B-1
LOW POWER BUS(MEASURED)
C
04236
849PAB81611
1 OF 4

POWER SYSTEM - BOARD 2

Drawing #PL849PABB1620

Item #	Part #	Description	MFG	QTY
1		T4 Aluminum	MMC	1
2		2/56 Screw	NL Fasteners	4
3	Module 38			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	NH-5,270ohm	1%, 50W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1
4	Module 37			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	H-5,50 ohm	1%, 50W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1
5	Module 36			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5,200ohm	1%, 5W Resistor	Dale	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1

POWER SYSTEM - BOARD 2
Con't

Drawing #PL849PABB1620

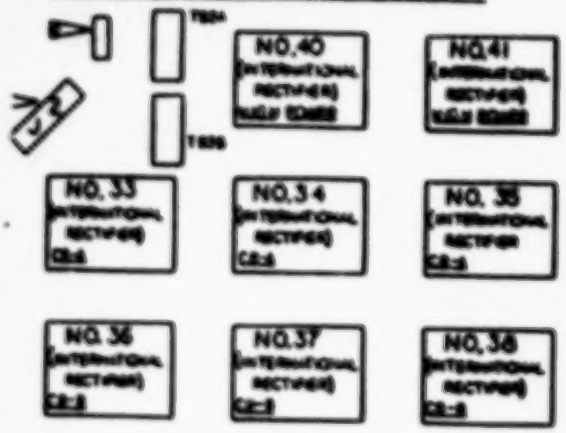
Item #	Part #	Description	MFG	QTY
6	Module 35			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	NH-50,27ohm	1%, 50W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1
7	Module 34			
	6321	Solid State Relay	Cryden	1
	RH-50,50 ohm	1%, 50W Resistor	Dale	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1
8	Module 33			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-10,100ohm	1%, 10W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1

POWER SYSTEM - BOARD 2
Con't

Drawing #PL849PABB1620

Item #	Part #	Description	MFG	QTY
9	Module 41			
	6321	Solid State Relay	Cryden	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-25,16 ohm	1%, 25W Resistor	Dale	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	1
	RS2027	1 Watt, 27 ohm, 1%	Dale	1
	RC07GF271J	1/4 Watt, 270 ohm, 5%	Allen-Bradley	1
		Capton Tape		1
10	Module 40			
	6321	Solid State Relay		1
	NH-50,14 ohm	1%, 50W Resistor		2
	TVA-1305.5	20vF, 50V Capacitor	Sprague	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor		1
	2N2222A	Small Signal Transistor		1
	RS2027	1 Watt, 27 ohm, 1%		1
	RC07GF271J	1/4 Watt, 270 ohm, 5%		1
		Capton Tape		1
11	Measurement Network			
	RH-5, 1ohm	1%,5W Resistor	Dale	2
	RNC65J7501F	7.5K ohm, 1% Resistor	Dale	2
	RNC65H2262FS	22.6K ohm, 1% Resistor	Dale	2
	300TX1305	20vF Capacitor	Sprague	2
12	72103	Terminal Blocks	Vernitron	2
13	DB25S	'D' Type, 25 Pin Female Connector	Ampherol	1

BOARD NO.2 MODULE LAYOUT



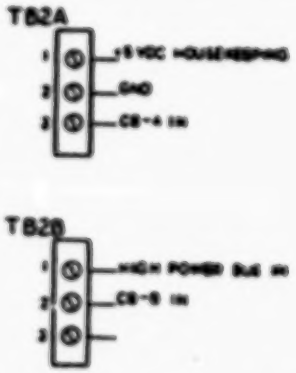
NOTES:

- CONNECT J2 TO A 25-PIN, 'D' TYPE CONNECTOR
- MEASUREMENT NETWORK

BOARD NO.2 J2 PIN DEFINITION

PIN NO.	FOR RELAY NO.	SIGNAL
1	40	ENK40
2	40	ENRS40*
3	40	END640
4	41	ENK41
5	41	ENRS41*
6	41	END641
7	33	ENK33
8	33	ENRS33*
9	33	END633
10	34	ENK34
11	34	ENRS34*
12	34	END634
14	35	ENK35
15	35	ENRS35*
16	35	END635
17	36	ENK36
18	36	ENRS36*
19	36	END636
20	37	ENK37
21	37	ENRS37*
22	37	END637
23	38	ENK38
24	38	ENRS38*
25	38	END638

BOARD NO. 2 TERMINAL BLOCK DESCRIPTION

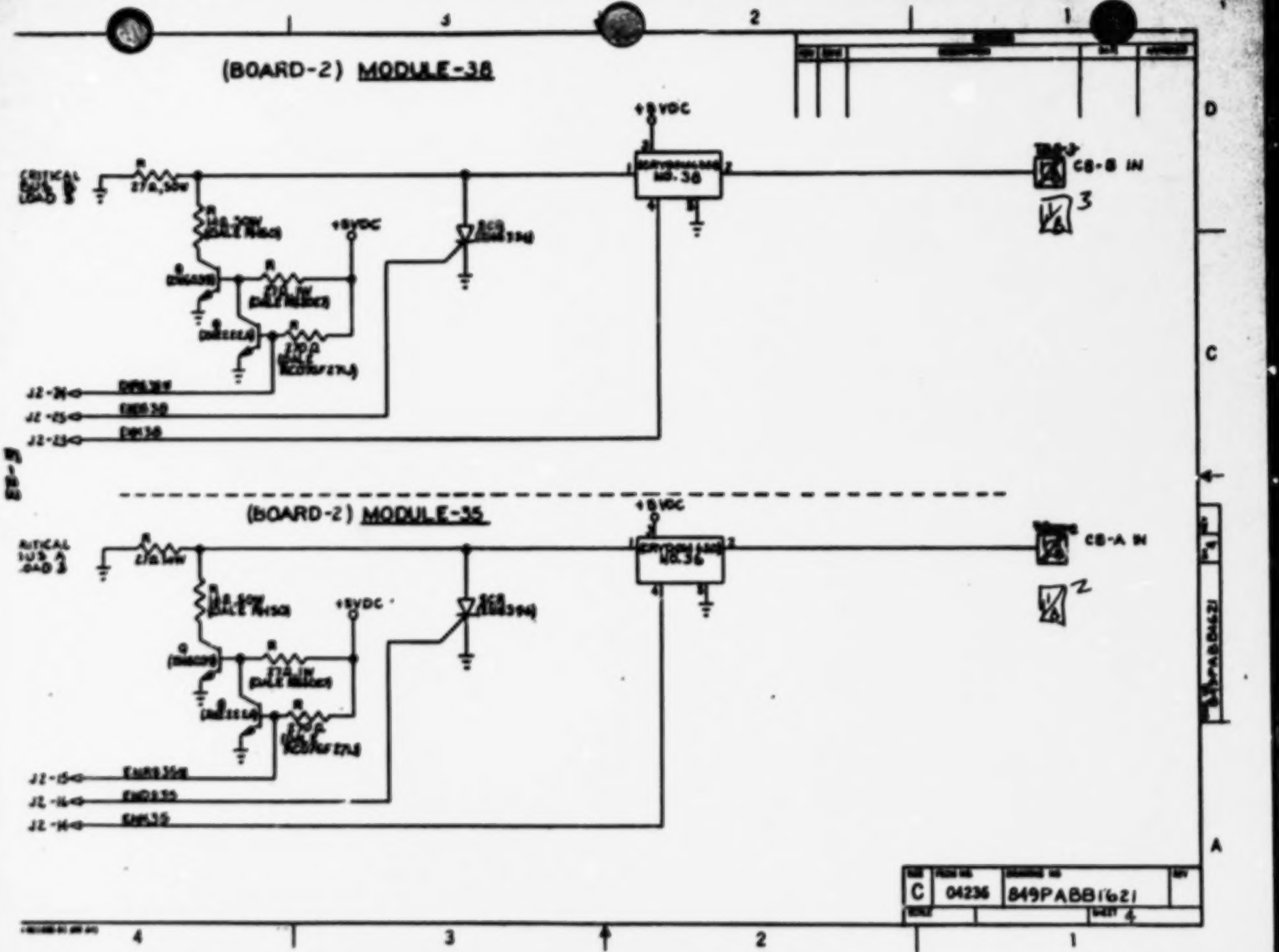


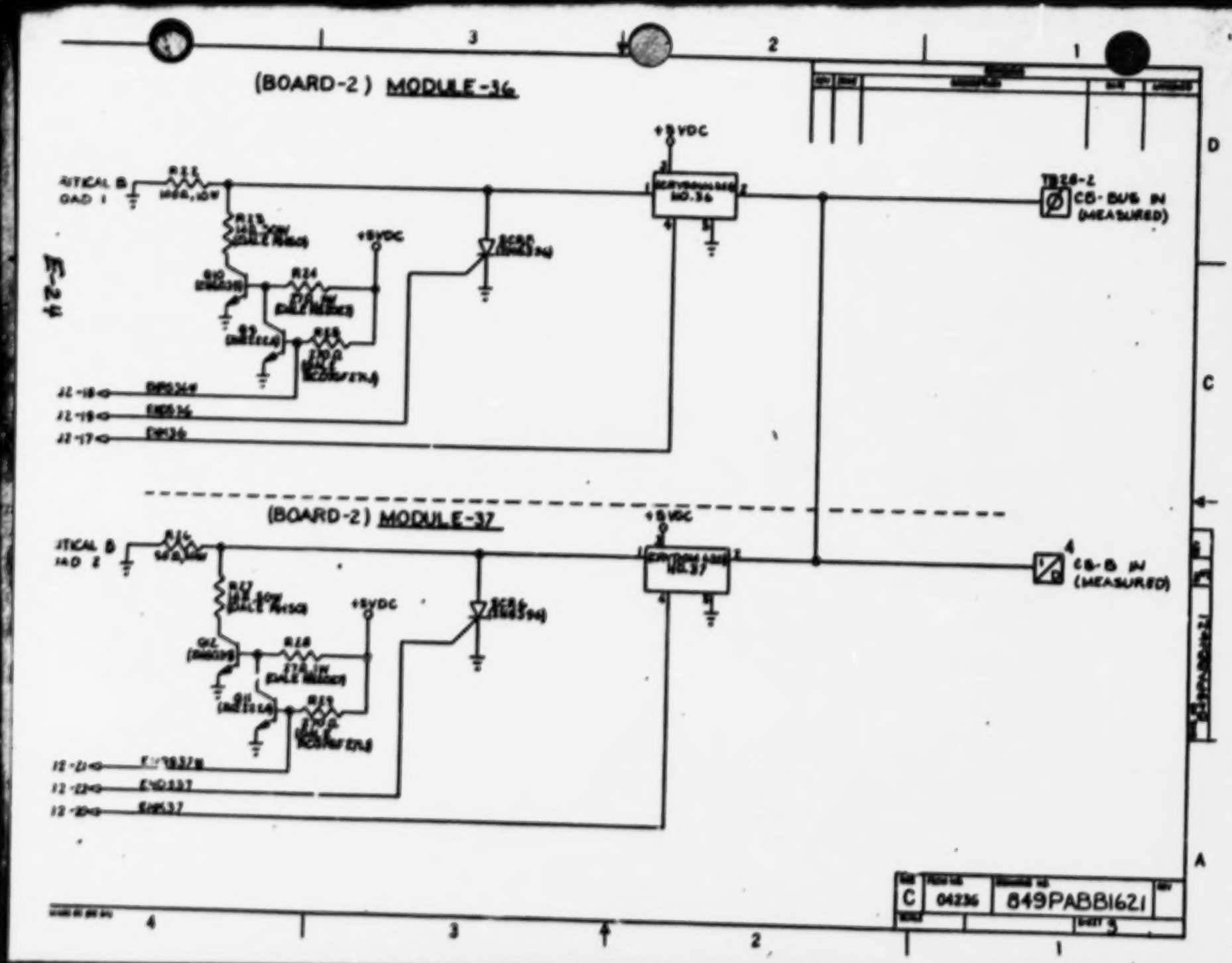
MARTIN MARIETTA CORPORATION

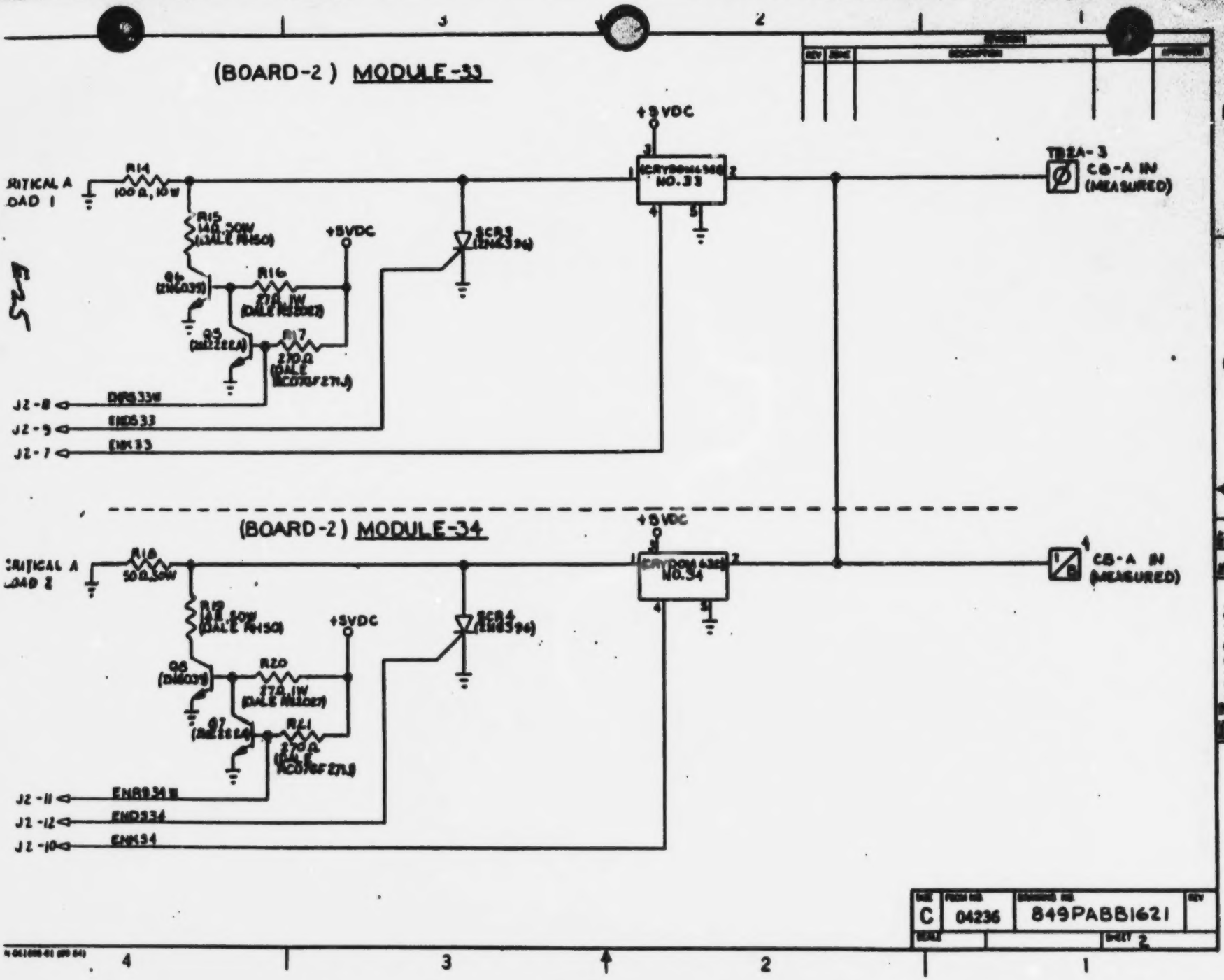
BOARD NO.2 LAYOUT AND SCHEMATIC, FIES PROGRAM

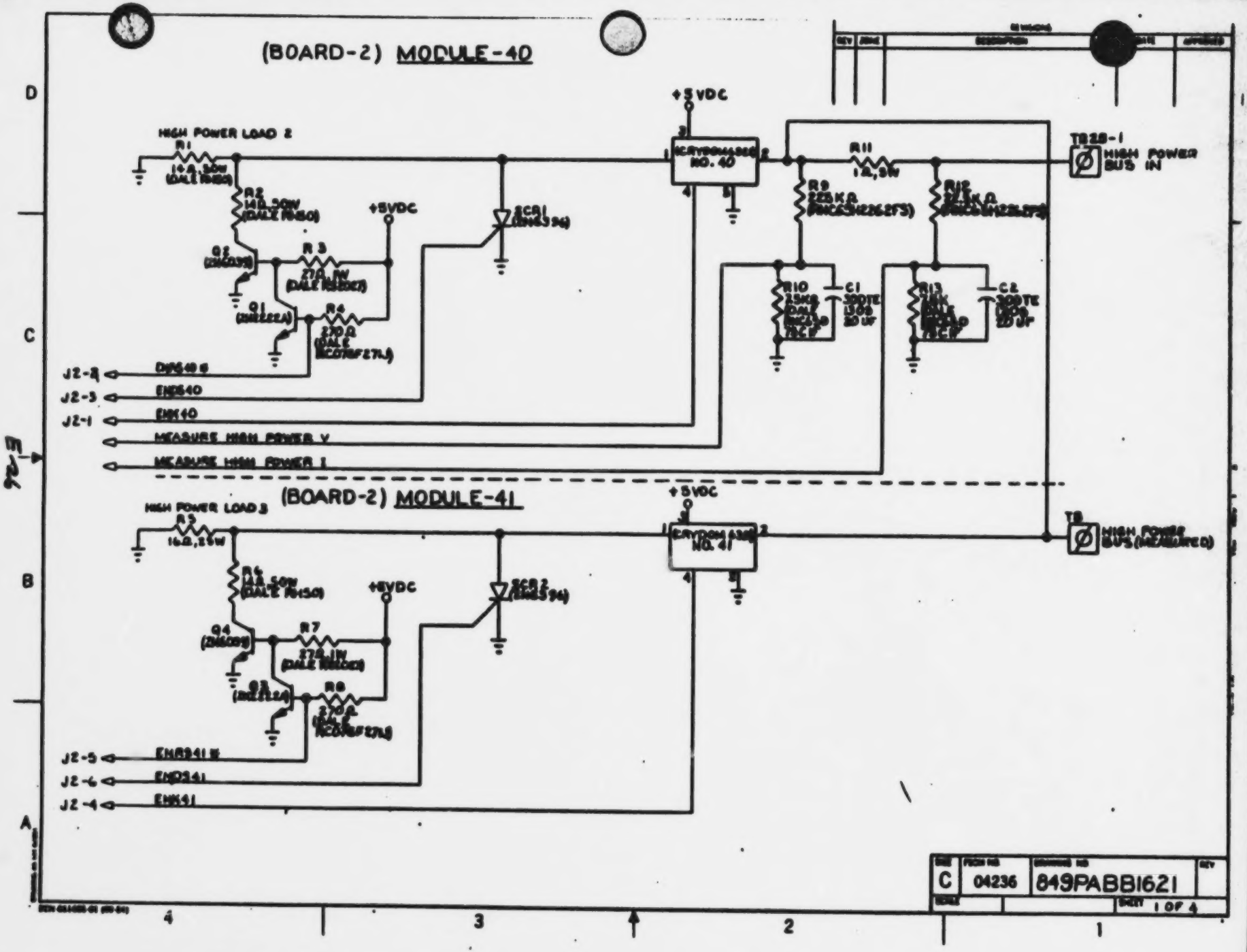
C 849PABB1620

203605


(BOARD-2) MODULE-38
+5VDC
CB-B IN
3
(BOARD-2) MODULE-35
+5VDC
CB-A IN
2
C
04236
849PABB1621


(BOARD-2) MODULE-36
+5VDC
MO.36
TB28-2
CB-BUS IN
(MEASURED)
(BOARD-2) MODULE-37
+5VDC
MO.37
CB-B IN
(MEASURED)
C
04236
849PABB1621


(BOARD-2) MODULE-33
+5VDC
CRITICAL A LOAD 1
R14 100Ω, 10W
R15 14Ω, 50W (DALE RH50)
Q6 (2N6039)
R16 27Ω, 1W (DALE RS2027)
Q5 (2N2222A)
R17 270Ω (DALE RC07GF271J)
+5VDC
SCR3 (2N6396)
NO.33
TB2A-3
CB-A IN (MEASURED)
J2-8 ENRS33W
J2-9 END533
J2-7 ENK33
(BOARD-2) MODULE-34
+5VDC
CRITICAL A LOAD 2
R18 50Ω, 50W
R19 14Ω, 50W (DALE RH50)
Q8 (2N6039)
R20 27Ω, 1W (DALE RS2027)
Q7 (2N2222A)
R21 270Ω (DALE RC07GF271J)
+5VDC
SCR4 (2N6396)
NO.34
CB-A IN (MEASURED)
J2-11 ENRS34W
J2-12 END534
J2-10 ENK34
C 04236 849PABB1621
SHEET 2


(BOARD-2) MODULE-40
HIGH POWER LOAD 2
R1
R2
R3
R4
Q1
Q2
+5VDC
SCR1
NO. 40
R11
R9
R12
R10
R13
C1
C2
TB28-1
HIGH POWER BUS IN
J2-2
J2-3
J2-1
END540
ENK40
MEASURE HIGH POWER V
MEASURE HIGH POWER I
(BOARD-2) MODULE-41
HIGH POWER LOAD 3
R5
R6
R7
R8
Q4
Q3
SCR2
NO. 41
TB
HIGH POWER BUS (MEASURED)
J2-5
J2-6
J2-4
END541
ENK41
C
04236
849PABB1621
1 OF 4

POWER SYSTEM - BOARD 3

Drawing #PL849PABB1630

Item #	Part #	Description	MFG	QTY
1		T4 Aluminum	MBC	1
2		2/56 Screw	NL Fasteners	4
3	Module 29			
	R10-E6-X2-V185	DPDT Relay	Potter-Brumfield	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RC07GF201J, 200ohm	1/4W, 5% Resistor	Allen-Bradley	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	2
	RS2027	1W, 27 ohm, 1%	Dale	1
	RC07GF271J	270 ohm, 5%, 1/4W	Allen-Bradley	2
	RH-5, .1 ohm	1%, 5W Resistor	Dale	1
	572-4820-01-05-16	Terminal Post	Midland-Ross	1
		Capton Tape		1
4	Module 28			
	R10-E6-X2-V185	DPDT Relay	Potter-Brumfield	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RC07GF201J, 200ohm	1/4W, 5% Resistor	Allen-Bradley	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	2
	RS2027	1W, 27 ohm, 1%	Dale	1
	RC07GF271J	270 ohm, 5%, 1/4W	Allen-Bradley	2
	RH-5, .1 ohm	1%, 5W Resistor	Dale	1
	572-4820-01-05-16	Terminal Post	Midland-Ross	1
		Capton Tape		1

PONR. STREEN - BOARD 3
Cont'd

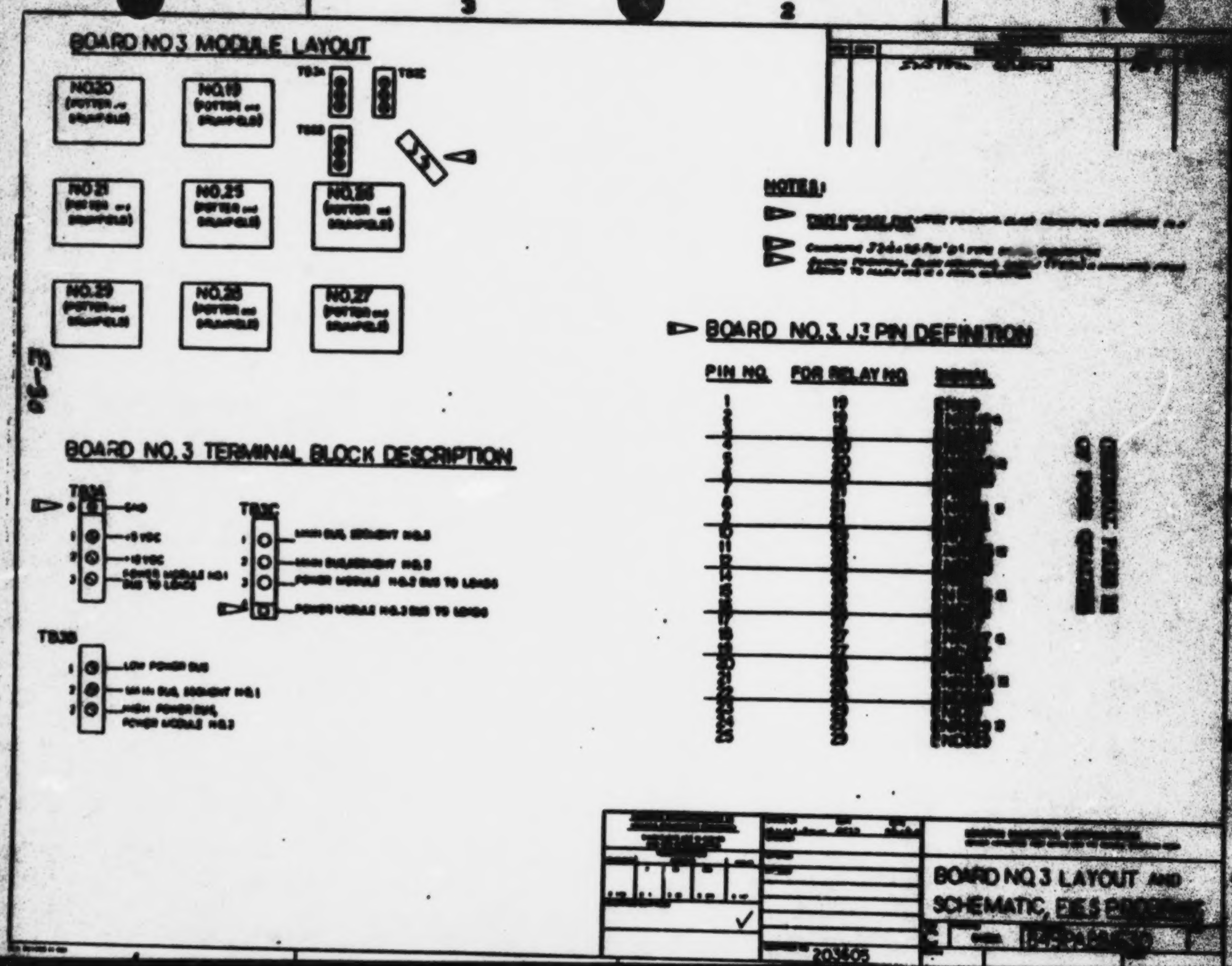
Drawing 8PL8498AB1630

Item #	Part #	Description	MFG	QTY
5	Module 27			
	R10-86-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RC07GF201J	1/4W, 5% Resistor	Allen-Bradley	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	2
	RS2027	1W, 27 ohm, Resistor	Dale	1
	RC07GF271J	270 ohm, 5%, 1/4W	Allen-Bradley	2
	572-4820-			
	01-05-16	Terminal Post	Midland-Ross	1
		Capton Tape		1
6	Module 26			
	R10-86-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RC07GF201J	1/4W, 5% Resistor	Allen-Bradley	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	2
	RS2027	1W, 27 ohm, Resistor	Dale	1
	RC07GF271J	270 ohm, 5%, 1/4W	Allen-Bradley	2
	572-4820-			
	01-05-16	Terminal Post	Midland-Ross	1
		Capton Tape		1
7	Module 25			
	R10-86-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RC07GF201J	1/4W, 5% Resistor	Allen-Bradley	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	2
	RS2027	1W, 27 ohm, Resistor	Dale	1
	RC07GF271J	270 ohm, 5%, 1/4W	Allen-Bradley	2
	572-4820-			
	01-05-16	Terminal Post	Midland-Ross	1
		Capton Tape		1

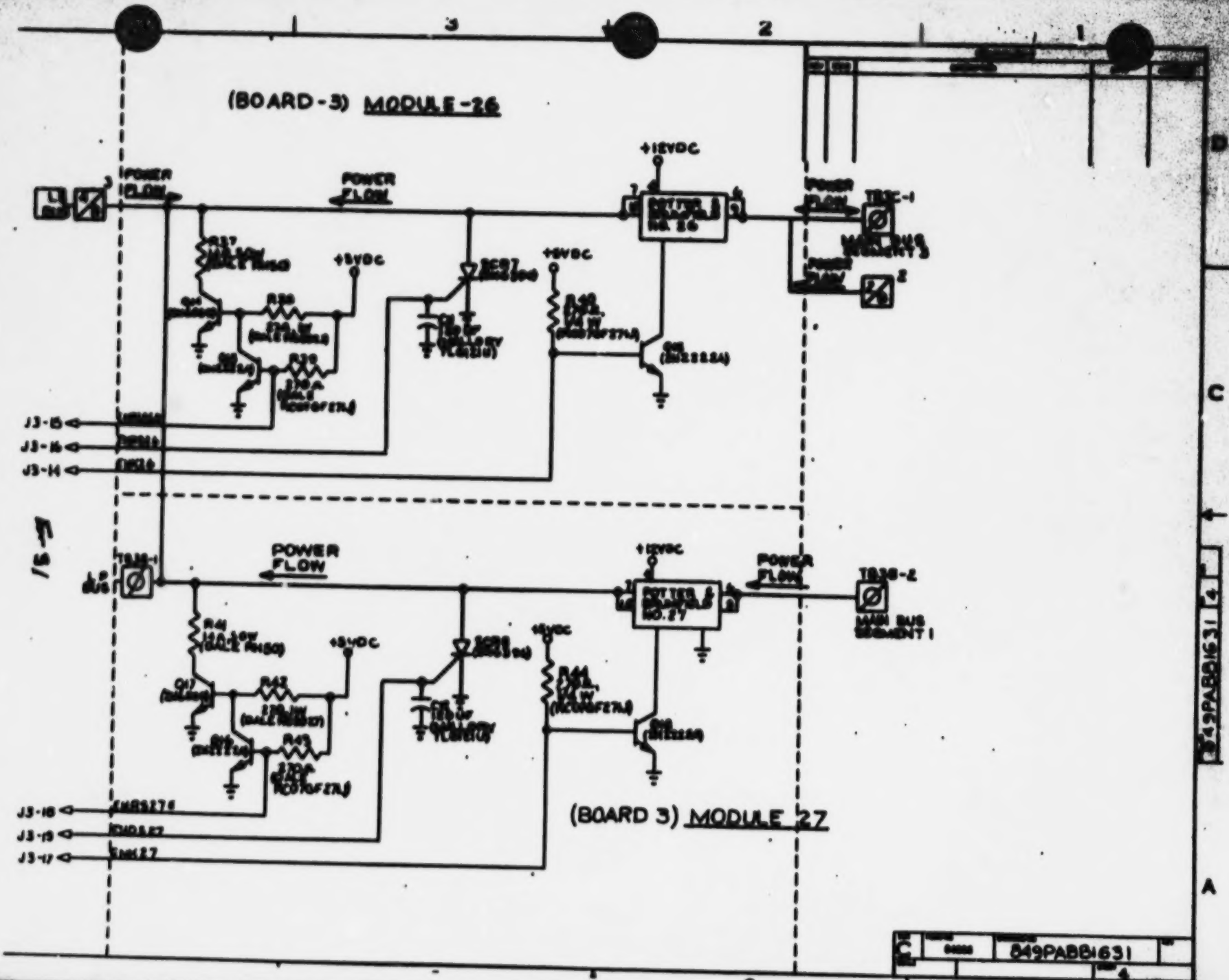
POWER SYSTEM - BOARD 3
Con't

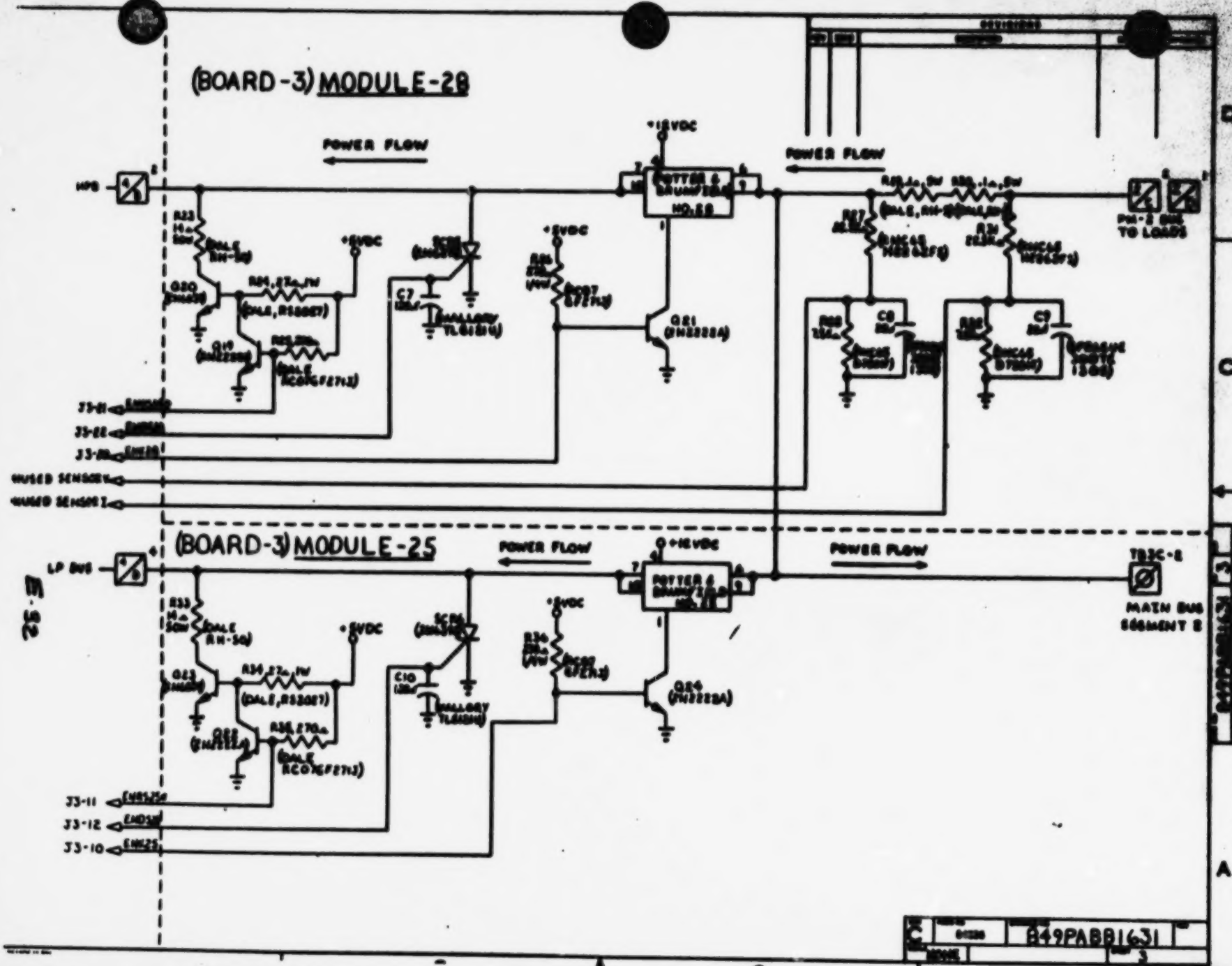
Drawing #PL849PABB1630

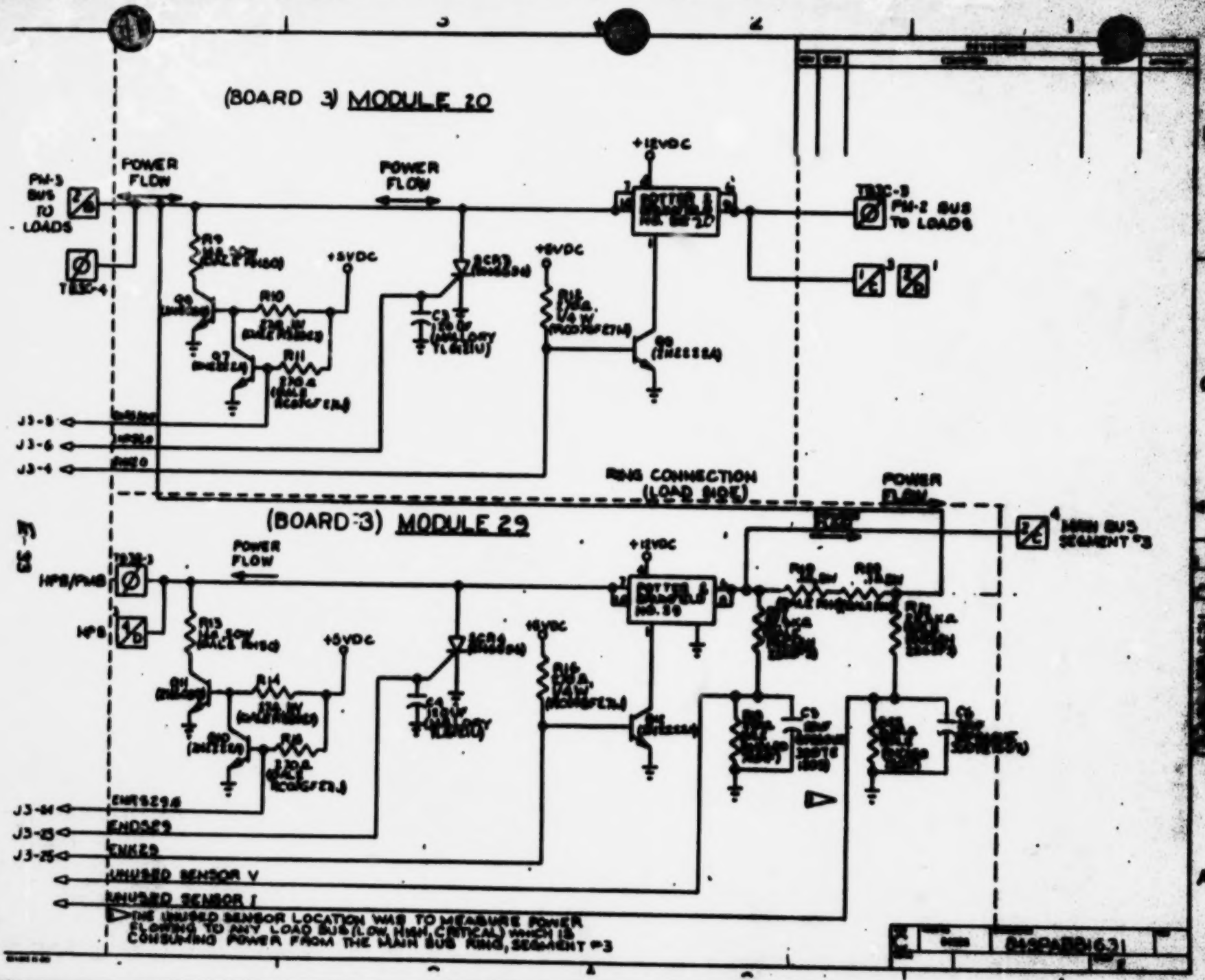
Item #	Part #	Description	MFG	QTY
8	Module 21			
	R10-B6-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RC07GF201J	1/4W, 5% Resistor	Allen-Bradley	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	2
	RS2027	1W, 27 ohm, Resistor	Dale	1
	RC07GF271J	270 ohm, 5%, 1/4W	Allen-Bradley	2
		Capton Tape		1
9	Module 19			
	R10-B6-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RC07GF201J	1/4W, 5% Resistor	Allen-Bradley	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	2
	RS2027	1W, 27 ohm, Resistor	Dale	1
	RC07GF271J	270 ohm, 5%, 1/4W	Allen-Bradley	2
		Capton Tape		1
10	Module 20			
	R10-B6-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	NH-50,14 ohm	1%, 50W Resistor	Dale	1
	RC07GF201J	1/4W, 5% Resistor	Allen-Bradley	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	2N2222A	Small Signal Transistor	Motorola	2
	RS2027	1W, 27 ohm, Resistor	Dale	1
	RC07GF271J	270 ohm, 5%, 1/4W	Allen-Bradley	2
		Capton Tape		1
11	72103	Terminal Blocks	Vernitron	3
12	DB25S	'D' Type, 25 Pin Connector, Female	Amphenol	1



BOARD NO 3 MODULE LAYOUT
NO.20 (POTTER and BRAMFIELD)
NO.19 (POTTER and BRAMFIELD)
NO.21 (POTTER and BRAMFIELD)
NO.25 (POTTER and BRAMFIELD)
NO.29 (POTTER and BRAMFIELD)
NO.28 (POTTER and BRAMFIELD)
NO.27 (POTTER and BRAMFIELD)
TB3C
NOTES:
BOARD NO. 3, J3 PIN DEFINITION
PIN NO.
FOR RELAY NO.
BOARD NO. 3 TERMINAL BLOCK DESCRIPTION
TB3A
GND
+5 VDC
+10 VDC
POWER MODULE NO.1 BUS TO LOADS
TB3C
POWER MODULE NO.2 BUS TO LOADS
POWER MODULE NO.3 BUS TO LOADS
TB3B
LOW POWER BUS
POWER MODULE NO.3
ORIGINAL PAGE IS OF POOR QUALITY
BOARD NO 3 LAYOUT AND SCHEMATIC, EIES PROGRAM
203605
E-30


(BOARD-3) MODULE-26
POWER FLOW
+12VDC
+5VDC
J3-15
J3-16
J3-14
J3-18
J3-19
J3-17
(BOARD 3) MODULE 27
MAIN BUS SEGMENT 1
849PABB1631


(BOARD-3) MODULE-28
POWER FLOW
+12VDC
POTTER & BRUMFIELD NO.28
POWER FLOW
HPB
+5VDC
+5VDC
Q21 (2N2222A)
PM-2 BUS TO LOADS
J3-21
J3-22
J3-20
(BOARD-3) MODULE-25
POWER FLOW
+12VDC
POTTER & BRUMFIELD NO.25
POWER FLOW
LP BUS
TB3C-2
MAIN BUS SEGMENT 2
+5VDC
+5VDC
Q24 (2N2222A)
J3-11
J3-12
J3-10
D
C
A
B49PABB1631


(BOARD 3) MODULE 20
POWER FLOW
POWER FLOW
PM-3 BUS TO LOADS
TB3C-4
+12VDC
+5VDC
+5VDC
TB3C-3
PM-2 BUS TO LOADS
J3-8
J3-6
J3-4
RING CONNECTION (LOAD SIDE)
POWER FLOW
(BOARD 3) MODULE 29
MAIN BUS SEGMENT #3
POWER FLOW
HPB/PMB
HPB
+12VDC
+5VDC
+5VDC
J3-24
J3-23
J3-25
ENDS29
UNUSED SENSOR V
UNUSED SENSOR I
THE UNUSED SENSOR LOCATION WAS TO MEASURE POWER FLOWING TO ANY LOAD BUS (LOW, HIGH, CRITICAL) WHICH IS CONSUMING POWER FROM THE MAIN BUS RING, SEGMENT #3

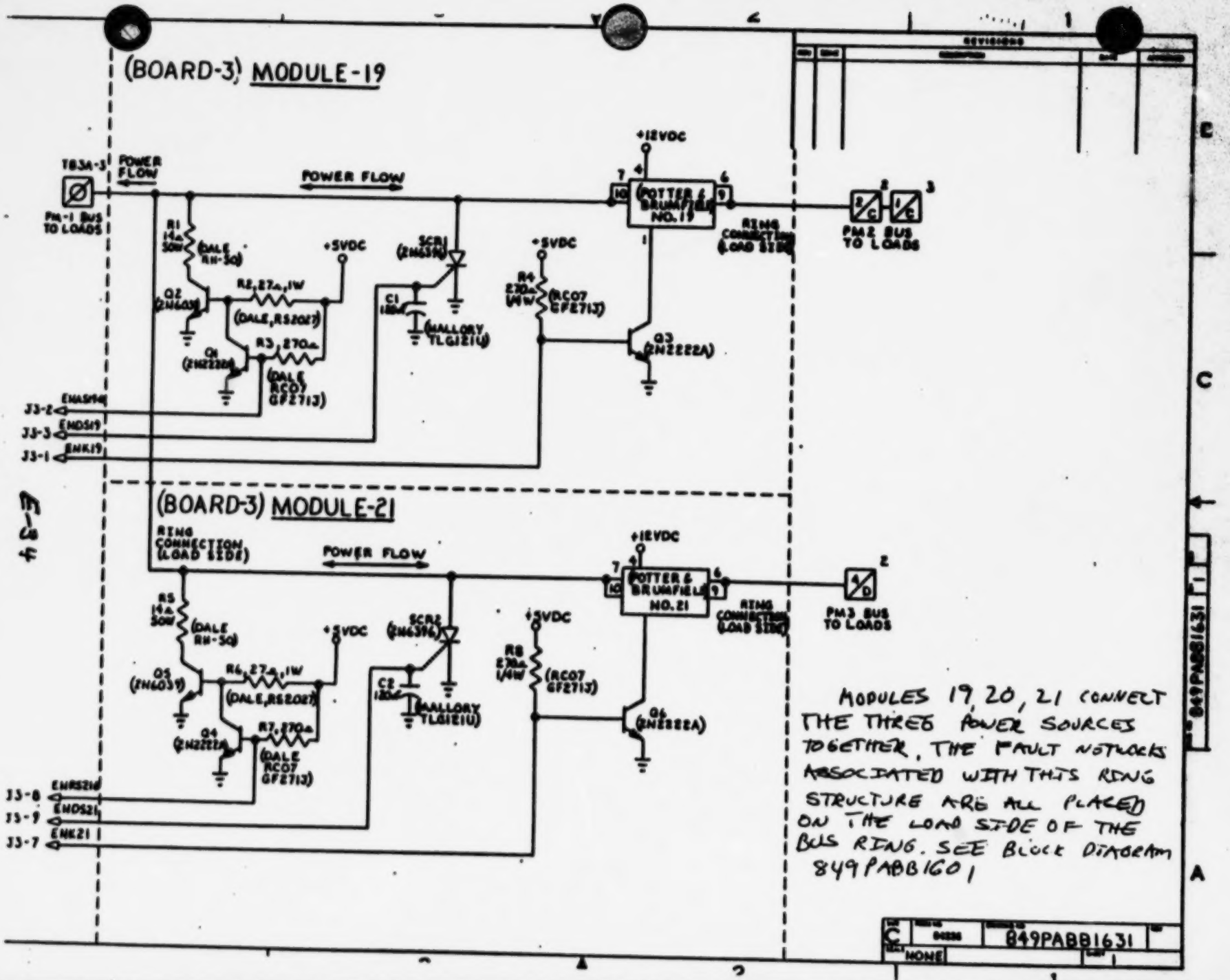

(BOARD-3) MODULE-19
TB3A-3
POWER FLOW
PM-1 BUS TO LOADS
POWER FLOW
R1 14Ω 50W (DALE RH-50)
Q2 (2N6039)
R2, 27Ω, 1W (DALE, RS2027)
Q1 (2N2222A)
R3, 270Ω (DALE RC07 GF2713)
+5VDC
SCR1 (2N6396)
C1 (MALLORY TL1211U)
+5VDC
R4 270Ω 1/4W (RC07 GF2713)
+12VDC
POTTER & BRUMFIELD NO. 19
Q3 (2N2222A)
RING CONNECTION (LOAD SIDE)
PM2 BUS TO LOADS
J3-2 ENHS19
J3-3 ENDS19
J3-1 ENK19
(BOARD-3) MODULE-21
RING CONNECTION (LOAD SIDE)
POWER FLOW
R5 14Ω 50W (DALE RH-50)
Q5 (2N6039)
R6, 27Ω, 1W (DALE, RS2027)
Q4 (2N2222A)
R7, 270Ω (DALE RC07 GF2713)
+5VDC
SCR2 (2N6396)
C2 (MALLORY TL1211U)
+5VDC
R8 270Ω 1/4W (RC07 GF2713)
+12VDC
POTTER & BRUMFIELD NO. 21
Q6 (2N2222A)
RING CONNECTION (LOAD SIDE)
PM3 BUS TO LOADS
J3-8 ENHS21
J3-9 ENDS21
J3-7 ENK21
MODULES 19, 20, 21 CONNECT THE THREE POWER SOURCES TOGETHER. THE FAULT NETWORKS ASSOCIATED WITH THIS RING STRUCTURE ARE ALL PLACED ON THE LOAD SIDE OF THE BUS RING. SEE BLOCK DIAGRAM 849PABB1601
849PABB1631
NONE

Fig 4

POWER SYSTEM - BOARD 4

Drawing #PL849PABB1640

Item #	Part #	Description	MFG	QTY
1		T4 Aluminum	MMC	1
2		2/56 Screw	NL Fasteners	4
3	Module 15			
	6321	Solid State Relay	Cryden	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	TVA-1305.5	20vF Capacitor	Sprague	1
	RH-5, .1ohm	1%, 5W Resistor	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL6121U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1% Resistor	Dale	2
	RNC65D7501F	7.5K ohm, 1% Resistor	Dale	2
	30DTX1305	20vF Capacitor	Sprague	2
4	Module 17			
	R10-E6-X2-V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTX1305	20vF Capacitor	Sprague	2

POWER SYSTEM - BOARD 4
Con't

Drawing #PL849PABB1640

Item #	Part #	Description	MFG	QTY
5	Module 32			
	R10-E6-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2
6	Module 31			
	R10-E6-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2

POWER SYSTEM - BOARD 4
Con't

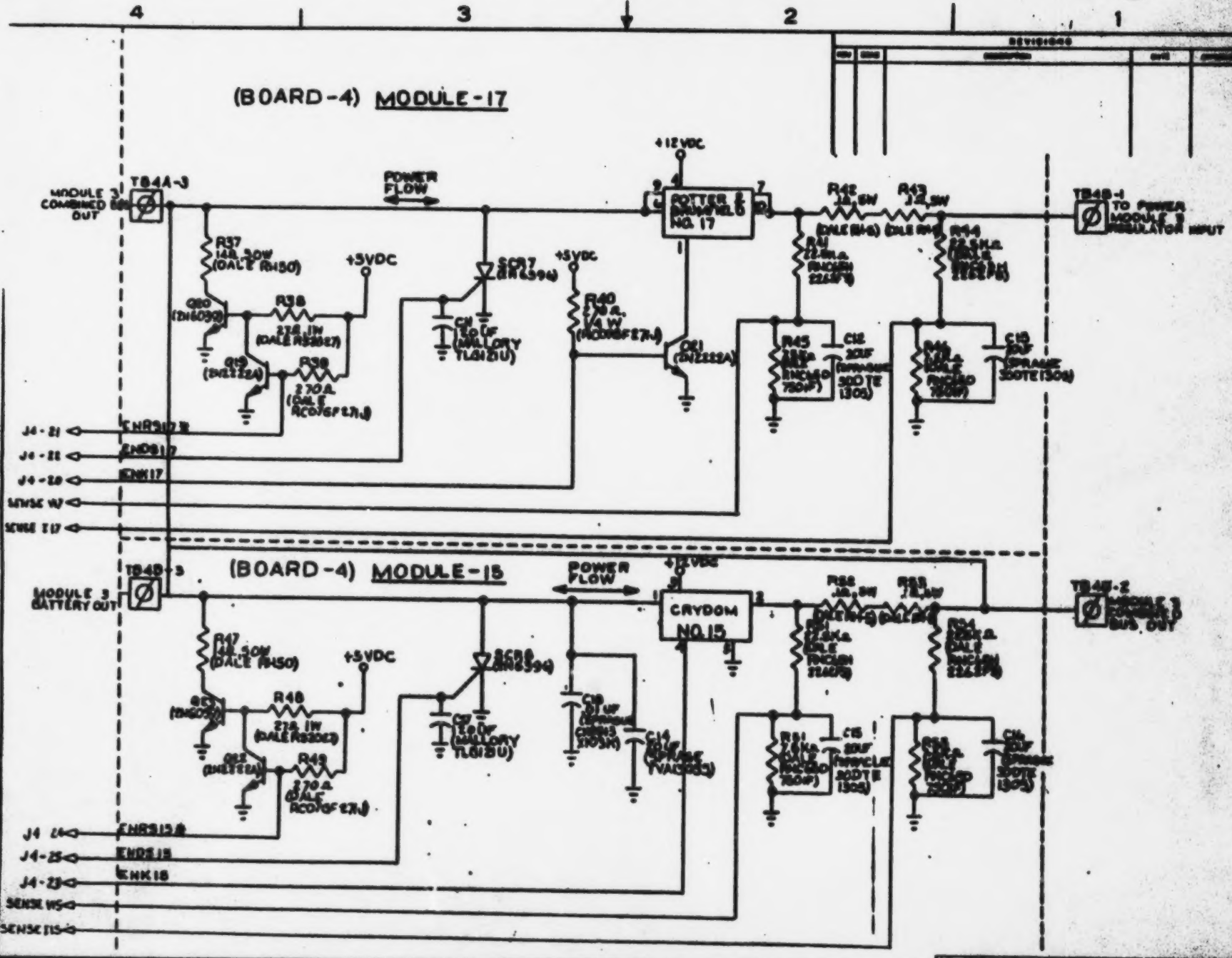
Drawing #PL2A9P6BB1640

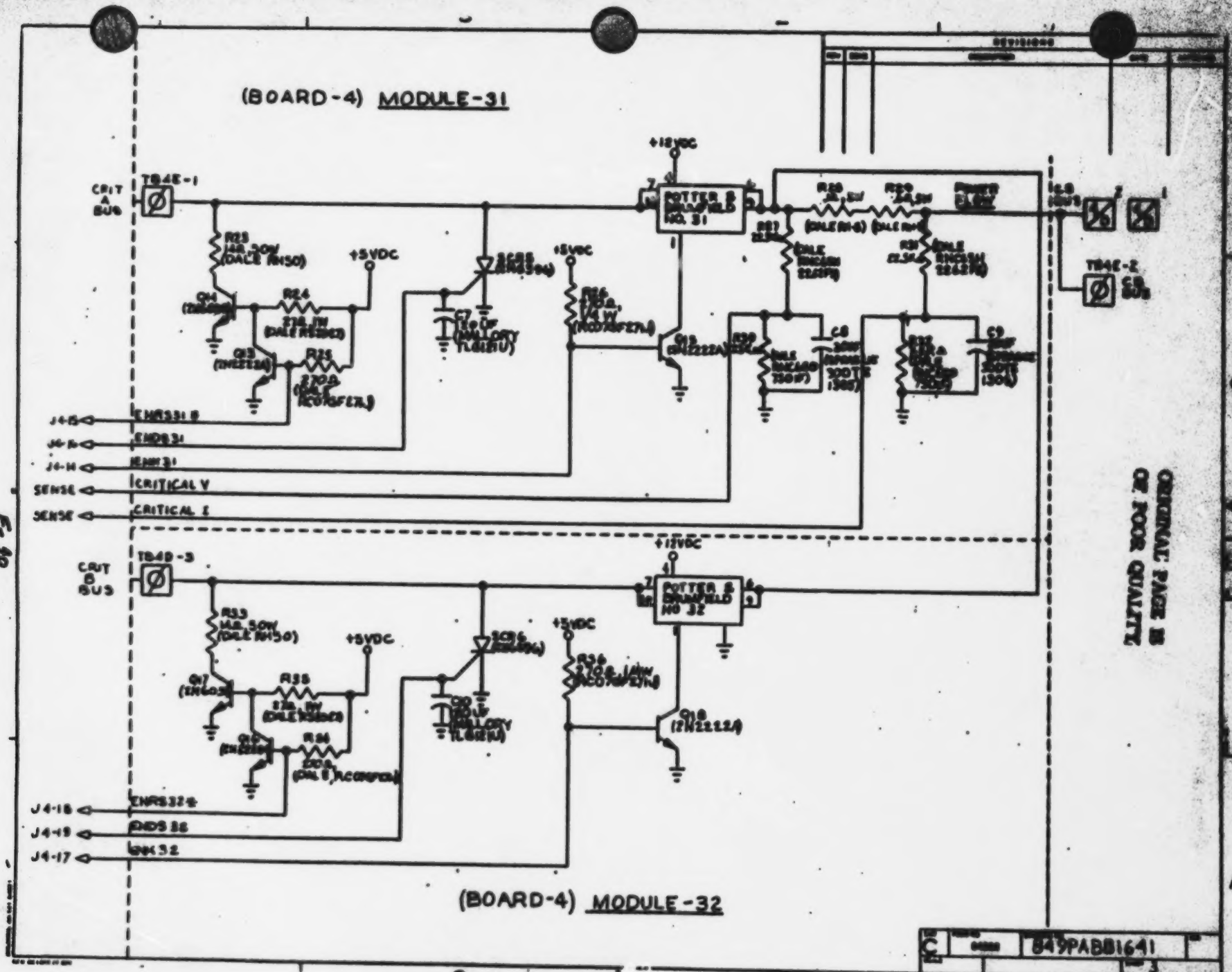
Item #	Part #	Description	MFG	QTY
7	Module 30			
	R10-E6-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2
8	Module 24			
	R10-E6-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2

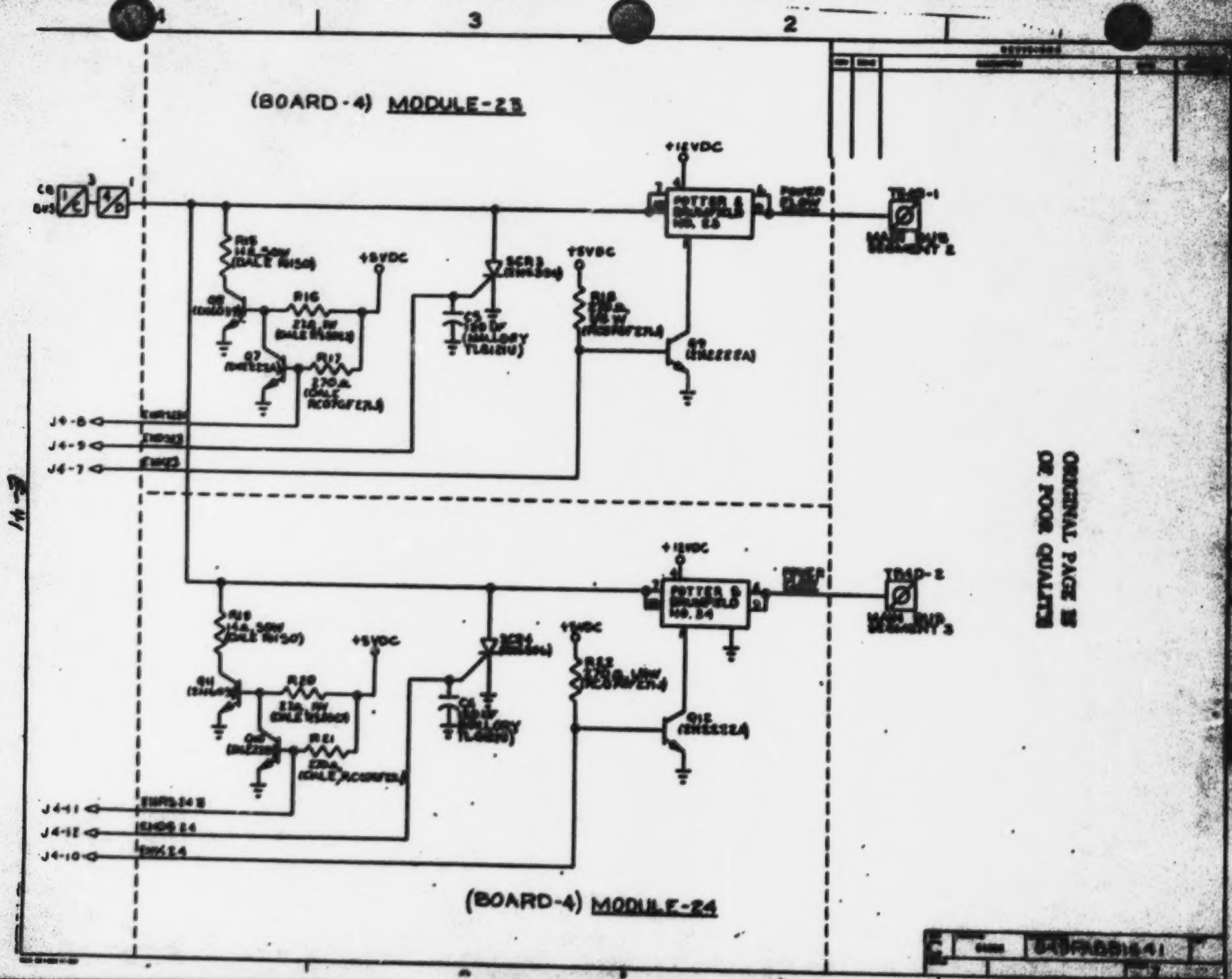
POWER SYSTEM - BOARD 4
Con't

Drawing #PL849PABB1640

Item #	Part #	Description	MFG	QTY
9	Module 23			
	R10-E6-X2-V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2
10	Module 22			
	R10-E6-X2-V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2

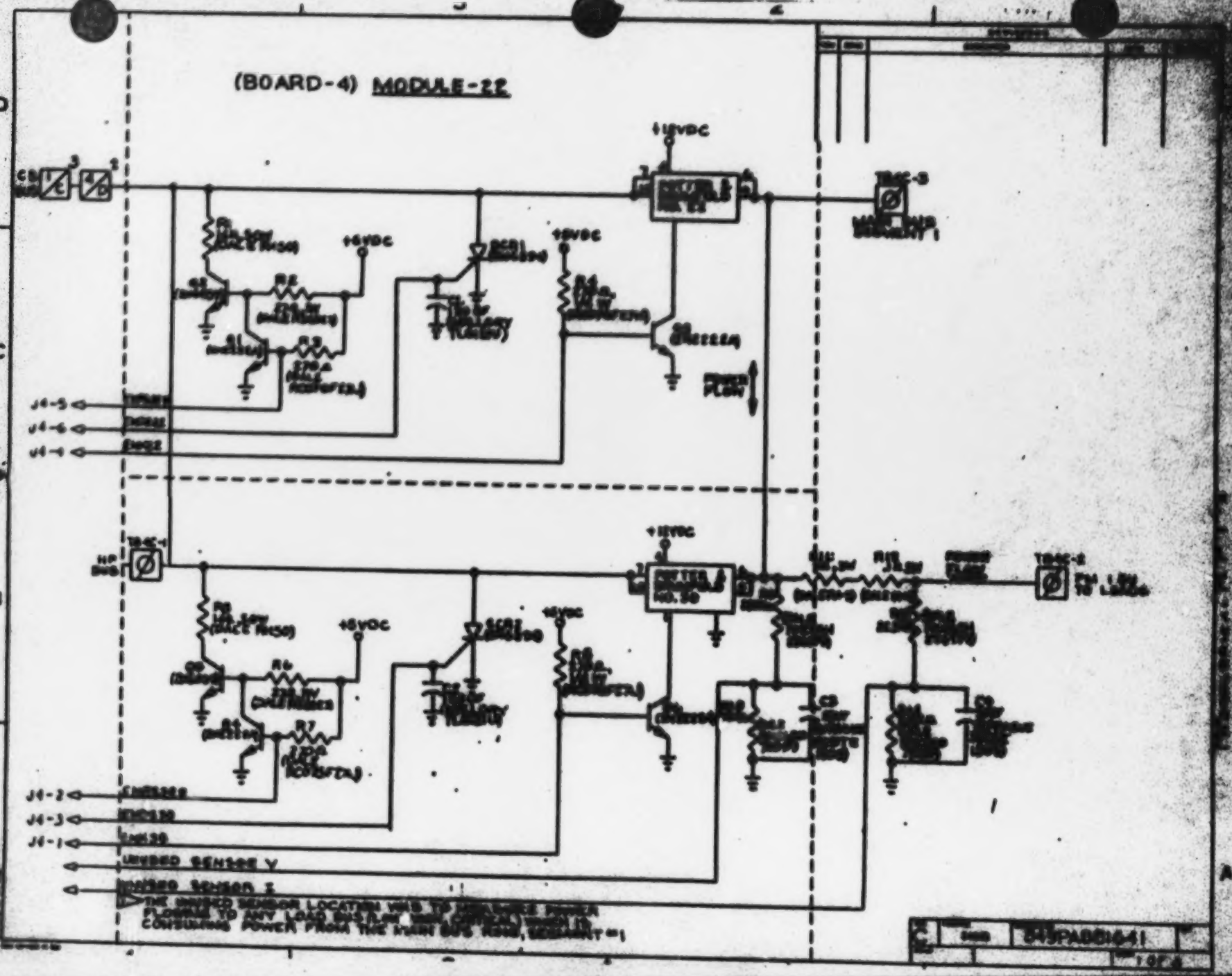

(BOARD-4) MODULE-17
MODULE 3 COMBINED OUT
TB4A-3
POWER FLOW
+12VDC
POTTER & BRINKERHOFF NO. 17
TB4B-1
TO POWER MODULE 3 REGULATOR INPUT
R37 1Ω, 5W (DALE RH50)
+5VDC
Q20 (2N6099)
R38 27Ω 1W (DALE RS20C)
Q19 (2N2222A)
R39 270Ω (DALE RC07GF271J)
SCR7 (2N6594)
C11 1.0UF (MALLORY TL612U)
R40 270Ω 1/4 W (RC07GF271J)
Q11 (2N2222A)
R42 1Ω, 5W (DALE RH5)
R43 1Ω, 5W (DALE RH5)
R41 22.6KΩ (RNC60 2262F)
R44 22.6KΩ (DALE RNC60 2262F)
R45 750Ω (RNC60 750F)
C12 20UF (SPRAGUE 30D TE 1305)
R46 750Ω (DALE RNC60 750F)
C13 20UF (SPRAGUE 30D TE 1305)
J4-21 ENRS17R
J4-22 ENDS17
J4-20 ENK17
SENSE W17
SENSE I17
(BOARD-4) MODULE-15
MODULE 3 BATTERY OUT
TB4B-3
POWER FLOW
+12VDC
CRYDOM NO. 15
TB4F-2
MODULE 3 BATTERY BUS OUT
R47 1Ω, 5W (DALE RH50)
+5VDC
Q21 (2N6099)
R48 27Ω 1W (DALE RS20C)
Q22 (2N2222A)
R49 270Ω (DALE RC07GF271J)
SCR8 (2N6594)
C17 1.0UF (MALLORY TL612U)
C16 .01UF (SPRAGUE CSR13 2103H)
C14 10UF (SPRAGUE TVA2223)
R50 1Ω, 5W (DALE RH5)
R53 1Ω, 5W (DALE RH5)
R51 22.6KΩ (DALE RNC60 2262F)
R54 22.6KΩ (DALE RNC60 2262F)
R52 750Ω (DALE RNC60 750F)
C15 20UF (SPRAGUE 30D TE 1305)
R55 750Ω (DALE RNC60 750F)
C16 20UF (SPRAGUE 30D TE 1305)
J4 24 ENRS15R
J4-25 ENDS15
J4-23 ENK15
SENSE W15
SENSE I15
4-37


(BOARD-4) MODULE-31
(BOARD-4) MODULE-32
CRIT A BUS
CRIT B BUS
TB4E-1
TB4E-2
TB4D-3
+12VDC
+5VDC
POTTER & BRUMFIELD NO. 31
POTTER & BRUMFIELD NO. 32
CRITICAL V
CRITICAL I
SENSE
J4-15
J4-16
J4-14
J4-18
J4-19
J4-17
ORIGINAL PAGE IS OF POOR QUALITY
849PAB81641

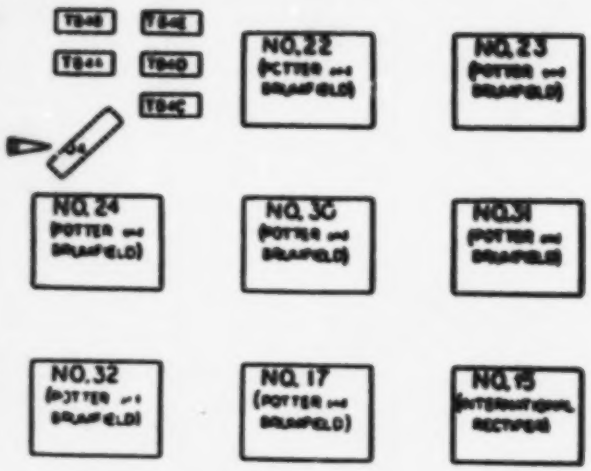


ORIGINAL PAGE IS
OF POOR QUALITY

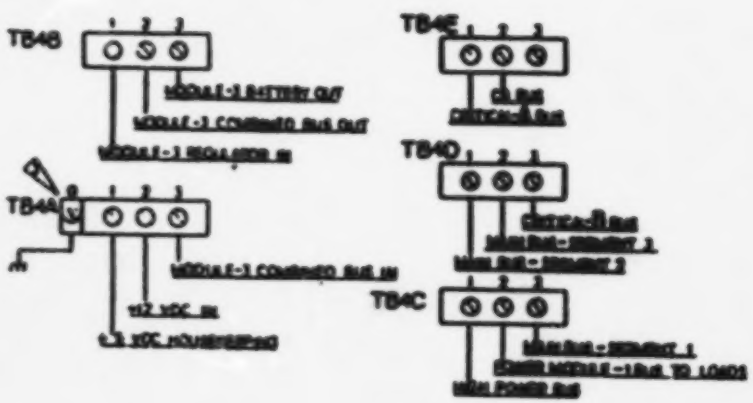
14-3


(BOARD-4) MODULE-22
+12VDC
+6VDC
+5VDC
J4-5
J4-6
J4-4
J4-2
J4-3
J4-1
UNUSED SENSOR Y
UNUSED SENSOR Z
POWER FLOW
THE UNUSED SENSOR LOCATION WAS TO MINIMIZE POWER FLOWING TO ANY LOAD BUS FLOW WHEN CRITICAL WHICH IS CONSUMING POWER FROM THE MAIN BUS RING, SEGMENT #1

BOARD NO.4 MODULE LAYOUT



BOARD NO.4 TERMINAL BLOCK DESCRIPTION



ORIGINAL PAGE IS OF POOR QUALITY

NOTES

1. NOT DRAWN TO SCALE
2. J4 IS A 25-PIN, D-TYPE SIGNAL CONNECTOR
3. TB4A, PIN 'D' IS A TERMINAL BLOCK MOUNT USED AS A GROUND RETURN

BOARD NO.4, J4 PIN DEFININITION

PIN NO.	FOR RELAY NO.	SIGNAL
1	30	ENK30
2	30	ENR30
3	30	ENR30
4	22	[illegible]
5	22	[illegible]
6	22	[illegible]
7	23	[illegible]
8	23	[illegible]
9	23	[illegible]
10	24	[illegible]
11	24	[illegible]
12	24	[illegible]
14	31	[illegible]
15	31	[illegible]
16	31	[illegible]
17	32	[illegible]
18	32	[illegible]
19	32	[illegible]
20	17	ENK17
21	17	ENR17
22	17	END17
23	15	ENK15
24	15	ENR15
25	15	END15

MARTIN MARIETTA CORPORATION

BOARD NO.4 SCHEMATIC AND LAYOUT- FIES PROGRAM

C 849FABB1640

203605

POWER SYSTEM - BOARD 5

Drawing #PL849PAB1650

Item #	Part #	Description	MFG	QTY
1		T4 Aluminum	MMC	1
2		2/56 Screw	ML Fasteners	4
3	Module 16			
	6321	Solid State Relay	Cryden	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	TVA-1305.5	20vF Capacitor	Sprague	1
	RH-5, .1ohm	1%, 5W Resistor	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL6121U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1% Resistor	Dale	2
	RNC65D7501F	7.5K ohm, 1% Resistor	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2
4	Module 18			
	R10-E6-X2-V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2

POWER SYSTEM - BOARD 5
Con't

Drawing #PL849PABB1650

Item #	Part #	Description	MFG	QTY
5	Module 12			
	R10-E6-X2-V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2
6	Module 14			
	6321	Solid State Relay	Cryden	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	TVA-1305.5	20vF Capacitor	Sprague	1
	RH-5, .1ohm	1%, 5W Resistor	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2

POWER SYSTEM - BOARD 5
Con't

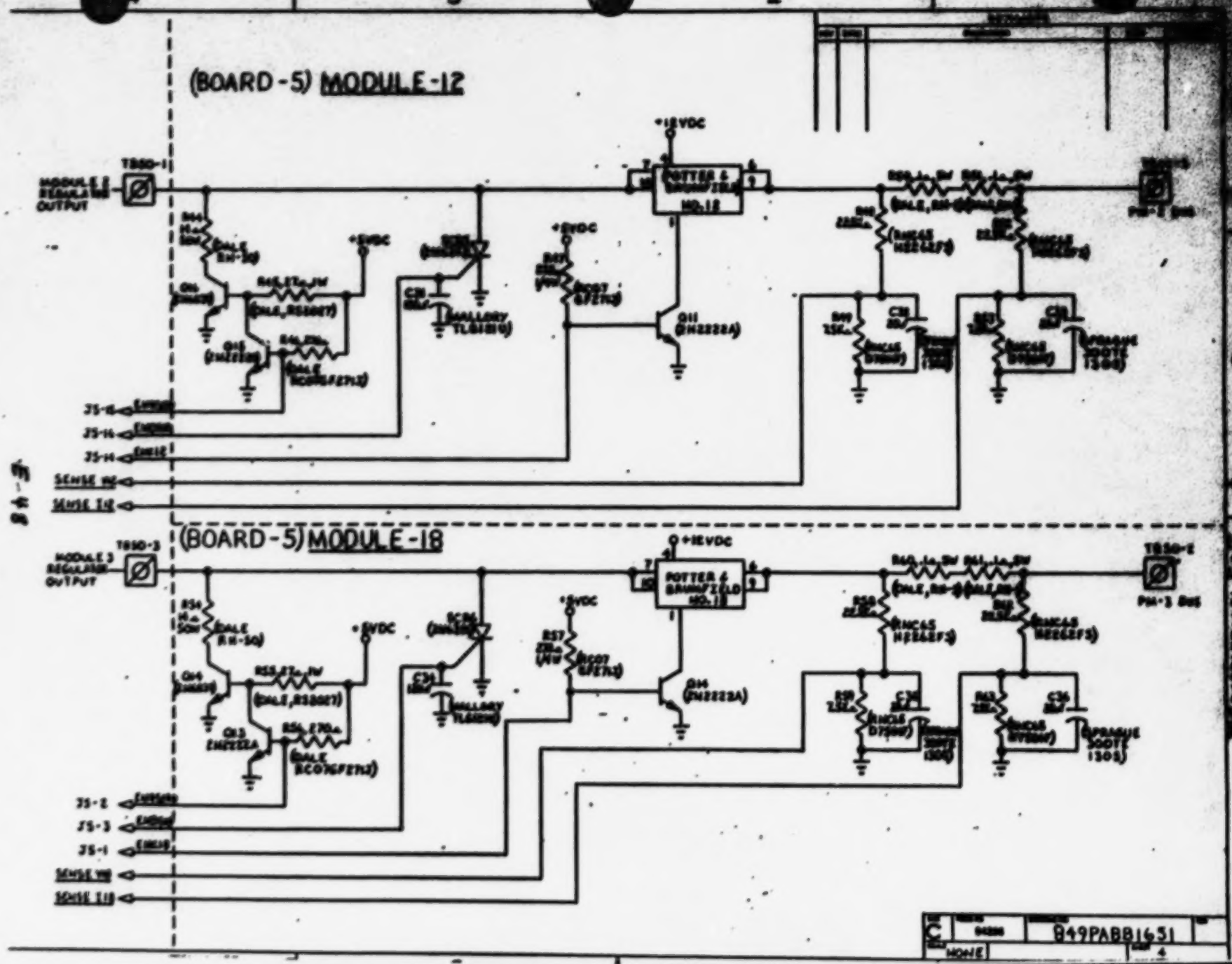
Drawing #PL849PAEB1650

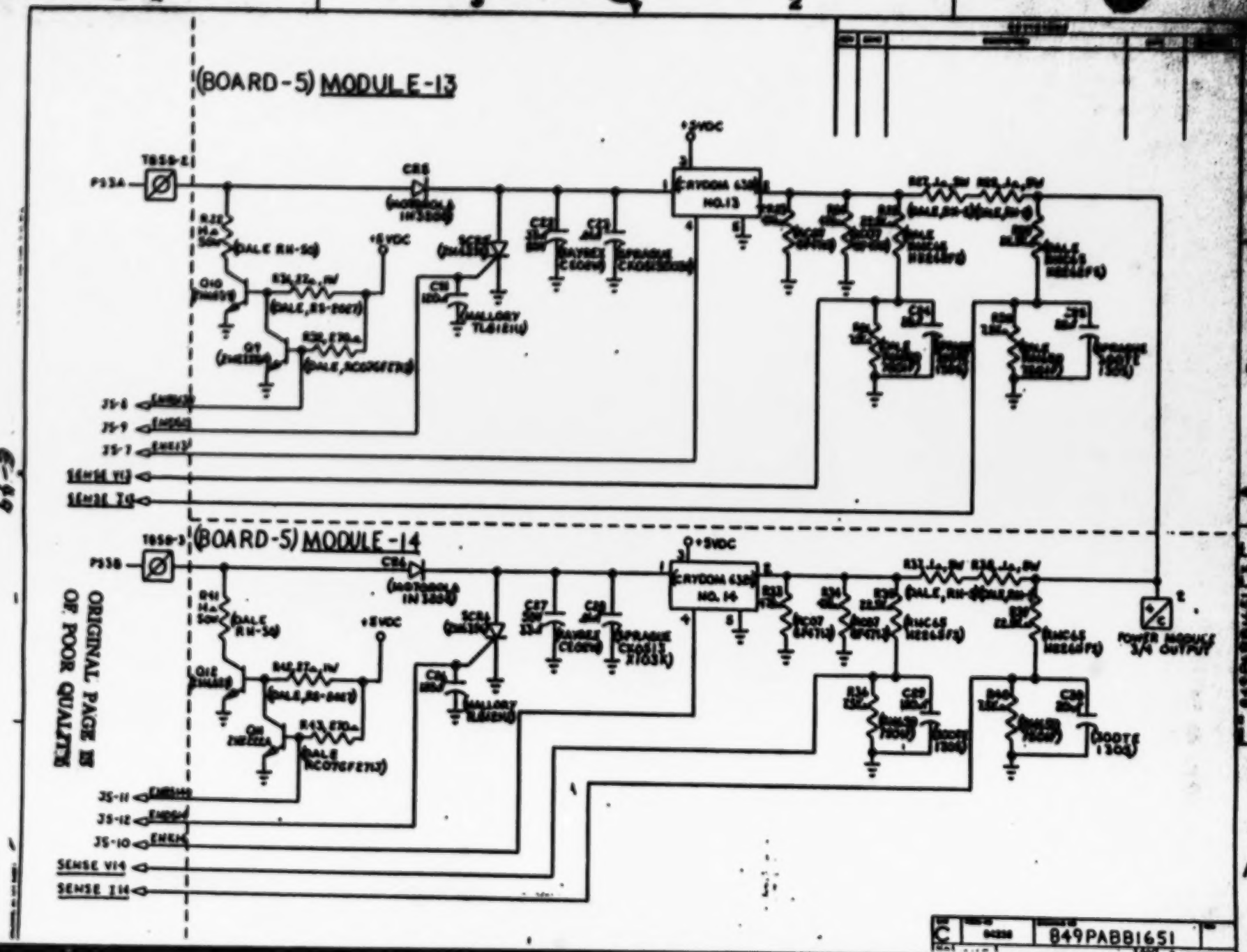
Item #	Part #	Description	MFG	QTY
7	Module 13			
	R10-E6-X2-V185	DPDT Relay	Potter-Brumfield	1
	RE-50,14 ohm	1%, 50W Resistor	Dale	1
	RE-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	300TX1305	20vF Capacitor	Sprague	2
8	Module 8			
	6321	Solid State Relay	Cryden	1
	RE-50,14 ohm	1%, 50W Resistor	Dale	1
	TVA-1305.5	20vF Capacitor	Sprague	1
	RE-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	300TX1305	20vF Capacitor	Sprague	2

POWER SYSTEM - BOARD 5
Con't

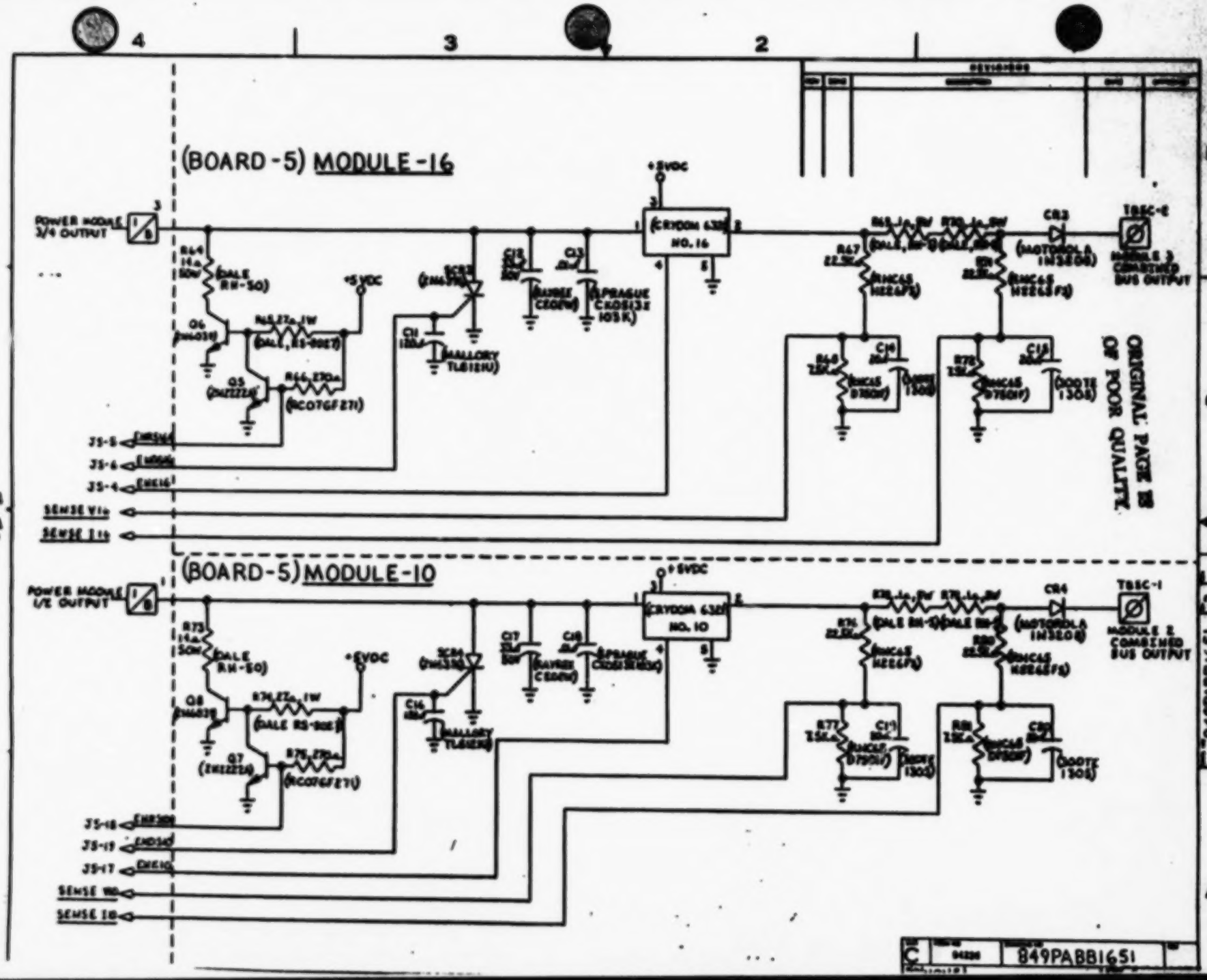
Drawing #PL049PABB1650

Item #	Part #	Description	MFG	QTY
9	Module 7			
	6321	Solid State Relay	Cryden	1
	RE-50,14 ohm	1%, 50W Resistor	Dale	1
	TVA-1305.5	20vF Capacitor	Sprague	1
	RE-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30D7K1305	20vF Capacitor	Sprague	2
10	Module 10			
	6321	Solid State Relay	Cryden	1
	RE-50,14 ohm	1%, 50W Resistor	Dale	1
	TVA-1305.5	20vF Capacitor	Sprague	1
	RE-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30D7K1305	20vF Capacitor	Sprague	2

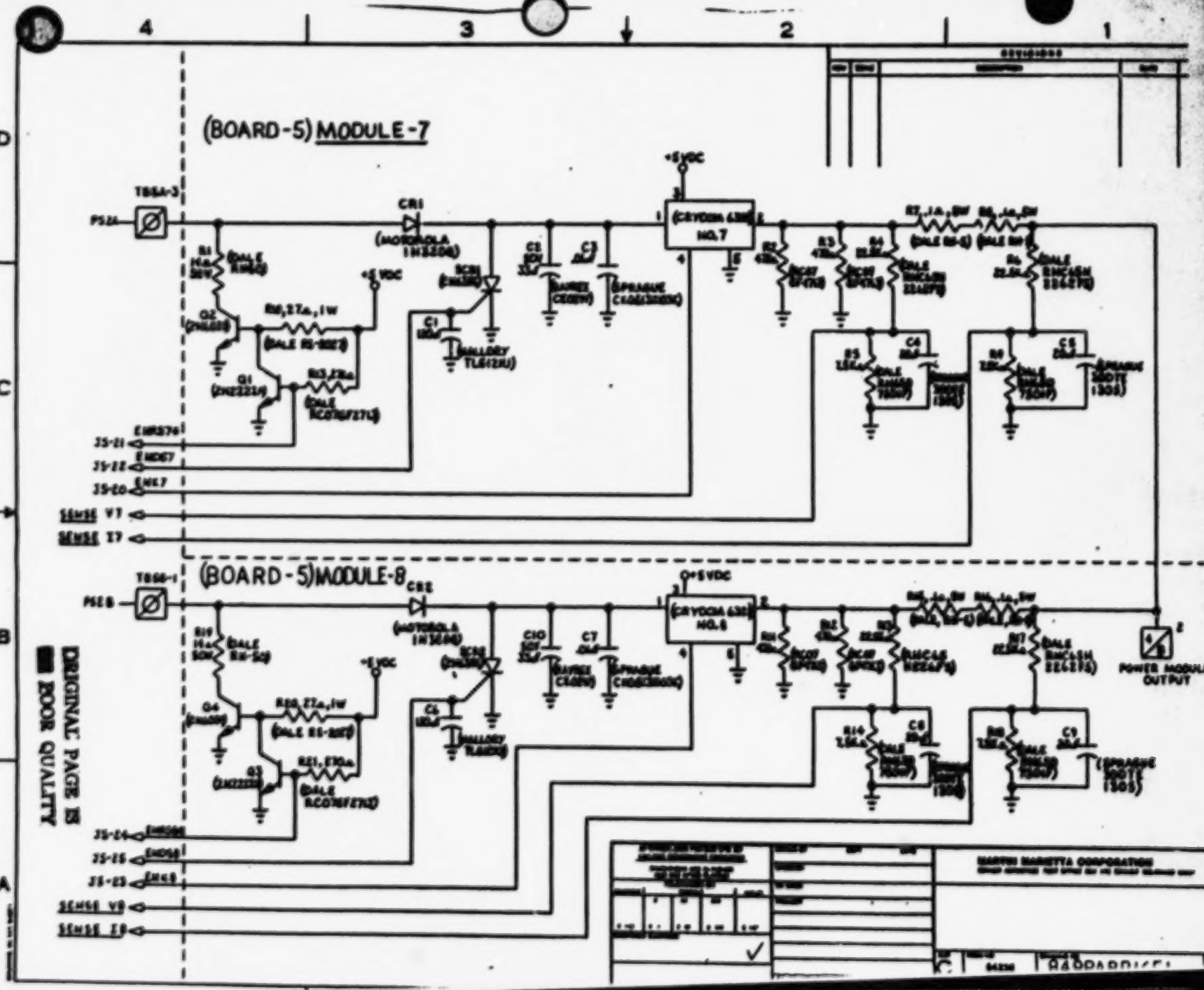

(BOARD-5) MODULE-12
(BOARD-5) MODULE-18
MODULE 2 REGULATOR OUTPUT
MODULE 3 REGULATOR OUTPUT
TB50-1
TB50-3
TB50-2
+12VDC
+5VDC
POTTER & BRUMFIELD NO.12
POTTER & BRUMFIELD NO.18
Q11 (2N2222A)
Q14 (2N2222A)
MALLORY TL8121U
SPRAGUE 300TE 1305
J5-15
J5-16
J5-14
J5-2
J5-3
J5-1
SENSE W2
SENSE T12
SENSE W8
SENSE T18
PM-2 BUS
PM-3 BUS
849PABB1651
NONE
4



ORIGINAL PAGE IS OF POOR QUALITY

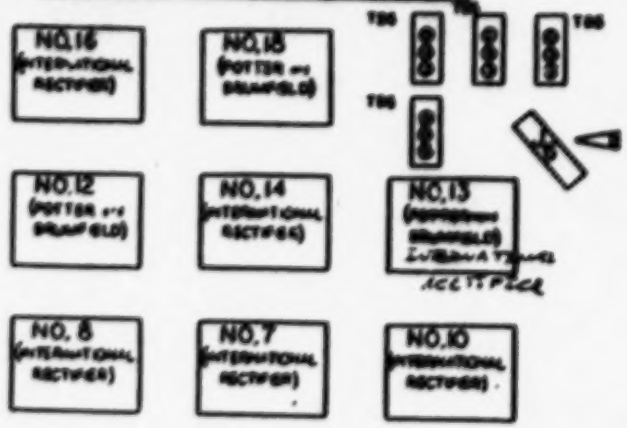


ORIGINAL PAGE IS OF POOR QUALITY

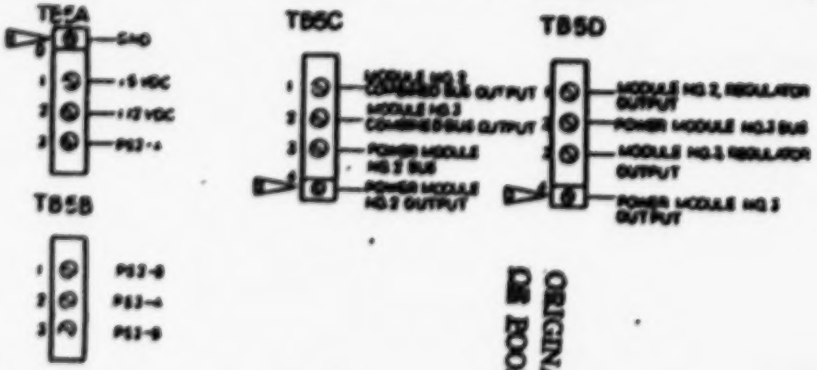


ORIGINAL PAGE IS
POOR QUALITY

BOARD NO.5 MODULE LAYOUT



BOARD NO.5 TERMINAL BLOCK DESCRIPTION



ORIGINAL PAGE IS OF POOR QUALITY

NOTES:

1. [illegible]
2. [illegible]
3. [illegible]

BOARD NO.5, J5 PIN DEFINITION

PIN NO.	FOR RELAY NO	SIGNAL
1	18	ENK18
2	18	ENRS18*
3	18	ENDS18
4	16	ENK16
5	16	ENRS16*
6	16	ENDS16
7	13	ENK13
8	13	ENRS13*
9	13	ENDS13
10	14	ENK14
11	14	ENRS14*
12	14	ENDS14
14	12	ENK12
15	12	ENRS12*
16	12	ENDS12
17	10	ENK10
18	10	ENRS10*
19	10	ENDS10
20	7	ENK7
21	7	ENRS7*
22	7	ENDS7
23	8	ENK8
24	8	ENRS8*
25	8	ENDS8

MARTIN MARIETTA CORPORATION

BOARD NO.5 LAYOUT AND SCHEMATIC, FIES PROGRA

C 94 PA881650

E-52

POWER SYSTEM - BOARD 6

Drawing #PL849PABB1660

Item #	Part #	Description	MFG	QTY
1		T4 Aluminum	MMC	1
2		2/56 Screw	NL Fasteners	4
3	Module 9			
	6321	Solid State Relay	Cryden	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	TVA-1305.5	20vF Capacitor	Sprague	1
	RH-5, .1ohm	1%, 5W Resistor	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	1
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL6121U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1% Resistor	Dale	2
	RNC65D7501F	7.5K ohm, 1% Resistor	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2
	LK0513X103K	.01 VF Capacitor	Sprague	1
4	Module 11			
	R10-E6-X2-V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2

POWER SYSTEM - BOARD 6
Con't

Drawing #PL849PA881660

Item #	Part #	Description	MFG	QTY
5	Module 6			
	R10-E6-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dalé	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2
6	Module 3			
	6321	Solid State Relay	Cryden	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1ohm	1%, 5W Resistor	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30DTE1305	20vF Capacitor	Sprague	2

POWER SYSTEM - BOARD 6
Cont'd

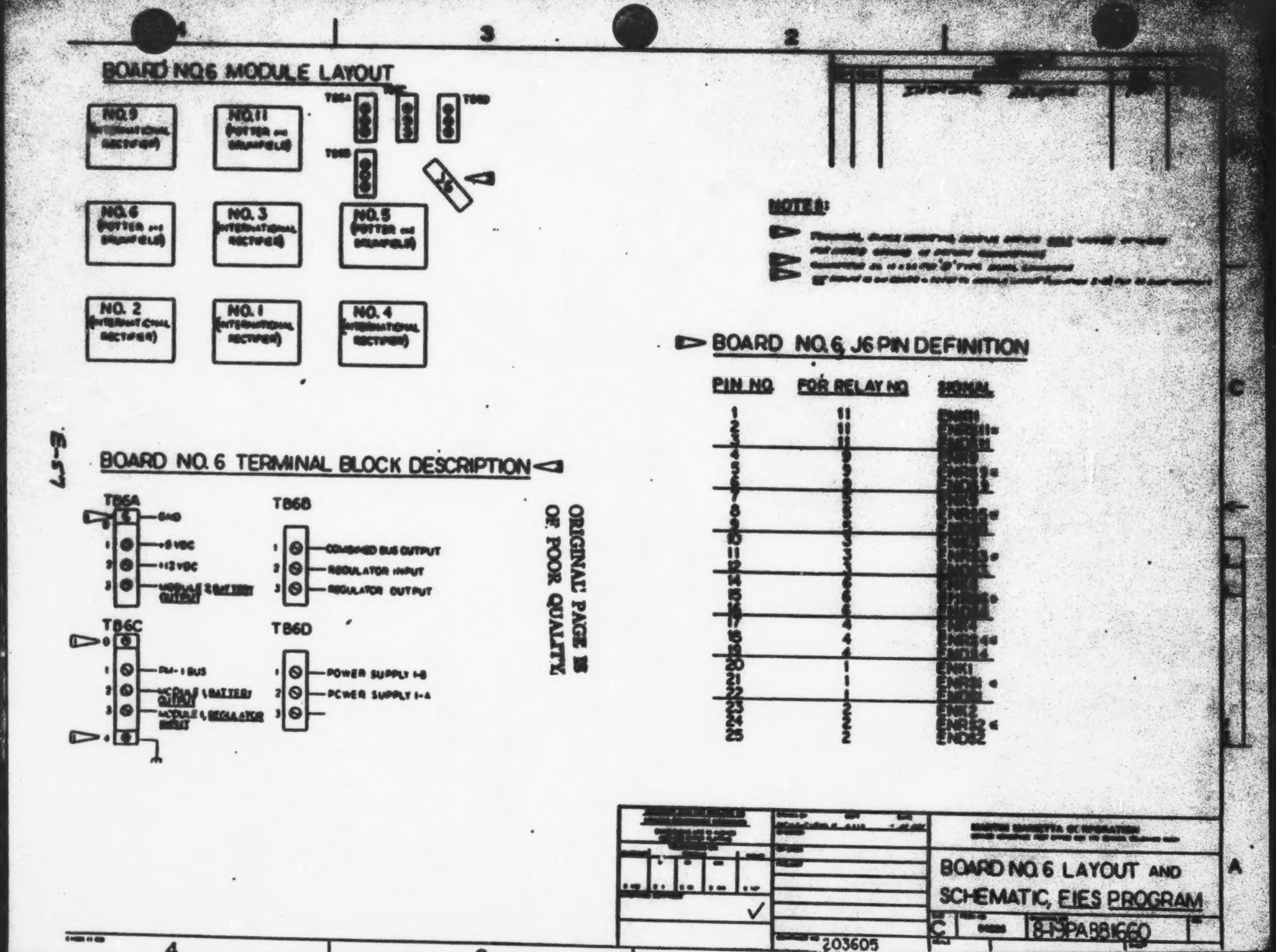
Drawing #PL849PA2B1660

Item #	Part #	Description	MFG	QTY
7	Module 5			
	R10-86-X2-			
	V185	DPDT Relay	Potter-Brumfield	1
	RN-50,14 ohm	1%, 50W Resistor	Dale	1
	RN-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30D7X1305	20vF Capacitor	Sprague	2
8	Module 2			
	6321	Solid State Relay	Cryden	1
	RN-50,14 ohm	1%, 50W Resistor	Dale	1
	RN-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N6396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30D7X1305	20vF Capacitor	Sprague	2
	RC07GF471J	470, 1/4W Resitor	Allen-Bradley	2
	TL6121U	120VF Capacitor	Mallory	1
	1N3208	Diode	Motorola	1
	LX0513X103K	.01VF Capacitor	Sprague	1

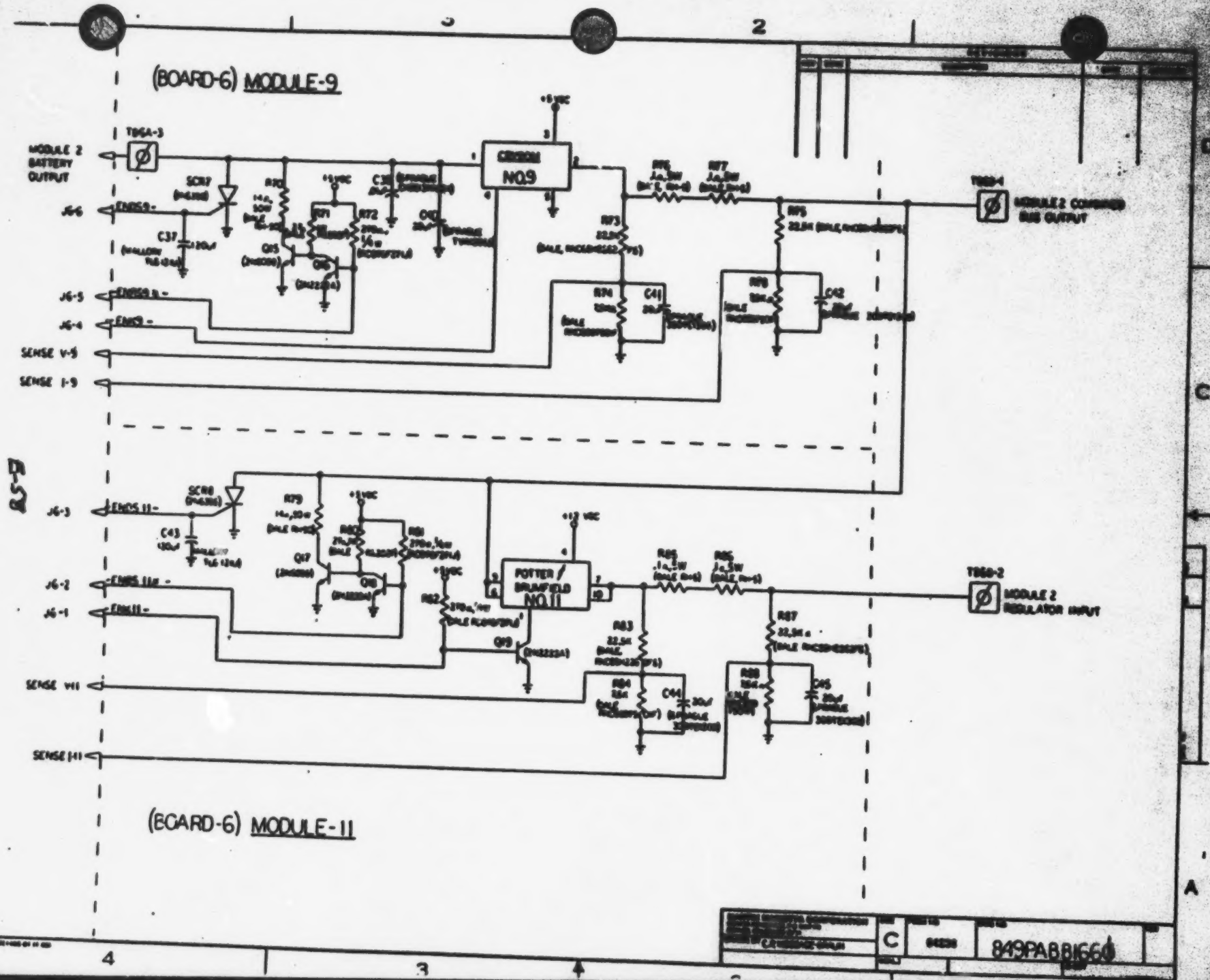
[illegible] - [illegible] 6
Cont'd

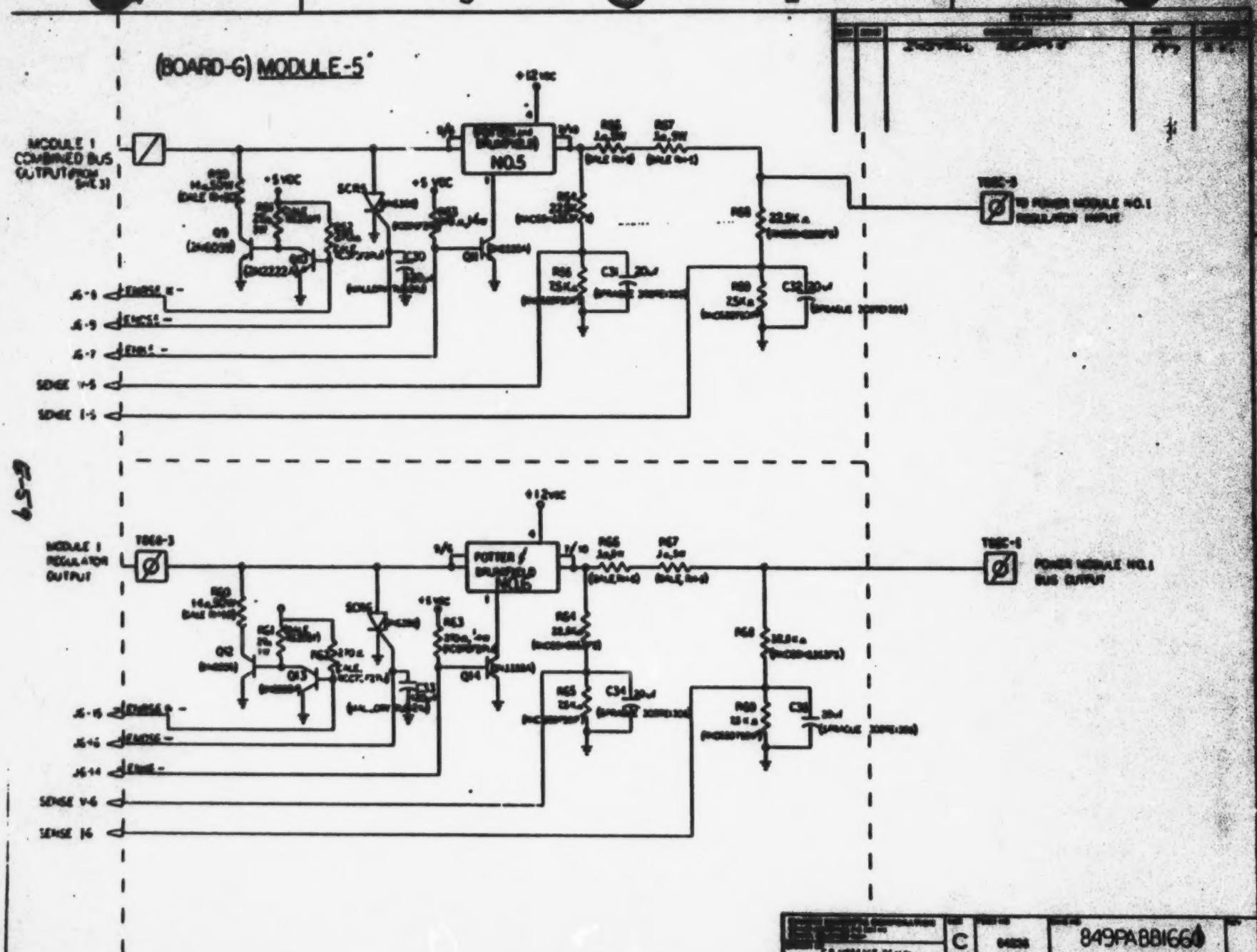
Drawing [illegible]

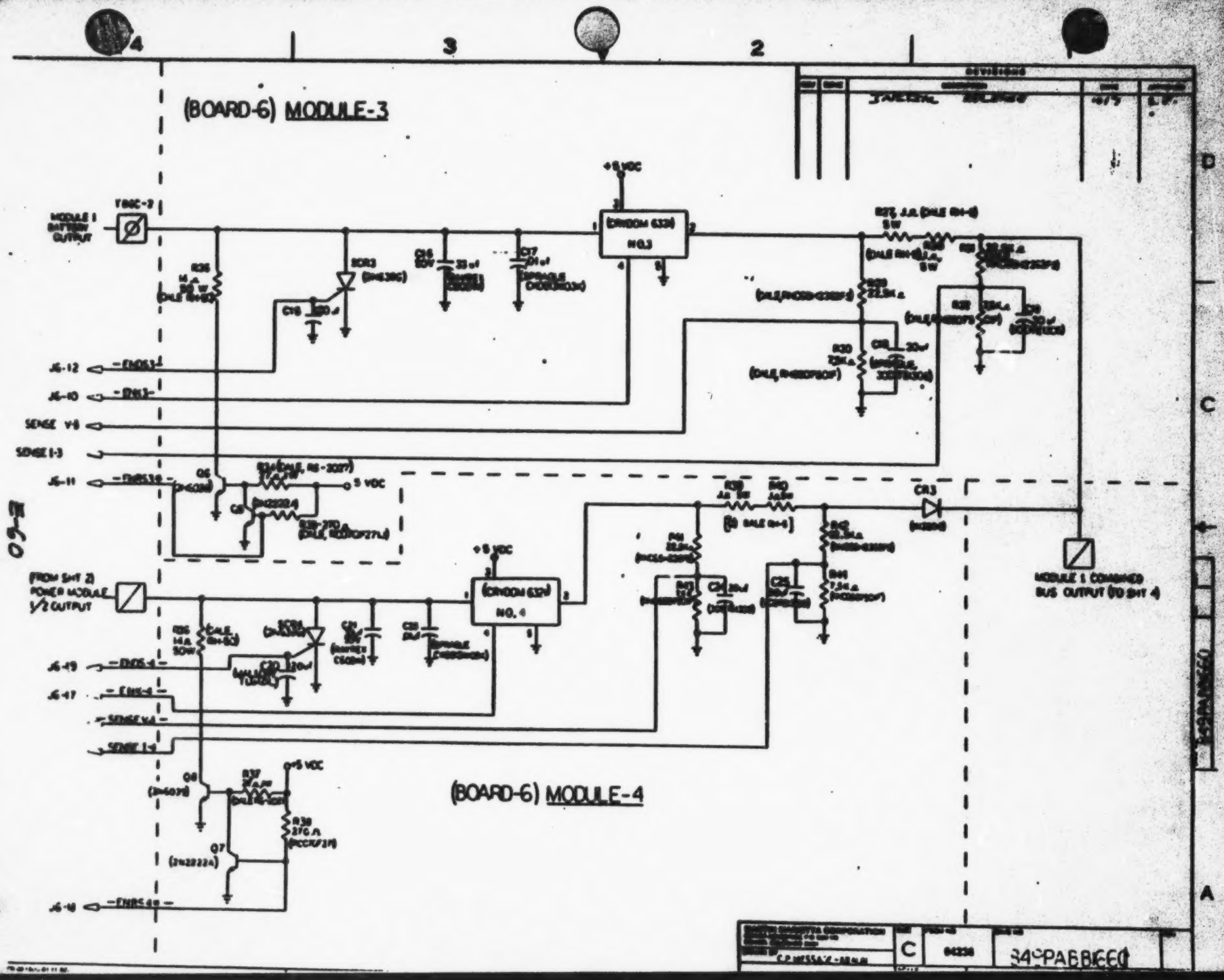
Item #	Part #	Description	MFG	QTY
9	Module 1			
	6321	Solid State Relay	Cryden	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N3396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	C[illegible]N	33vF Capacitor	RAYNEX	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30D[illegible]1305	20vF Capacitor	Sprague	2
	RC07GF471J	470, 1/4W Resitor	Allen-Bradley	2
	TL6121U	120VF Capacitor	Mallory	1
	1N3208	Diode	Motorola	1
	LX0513X103K	.01VF Capacitor	Sprague	1
10	Module 4			
	6321	Solid State Relay	Cryden	1
	RH-50,14 ohm	1%, 50W Resistor	Dale	1
	RH-5, .1 ohm	1% 5W Resisotr	Dale	2
	RC07GF271J	1/4W, 5% Resistor	Allen-Bradley	1
	RS2027	1W, 27 ohm, 1%	Dale	1
	2N2222A	Small Signal Transistor	Motorola	2
	2N3396	SCR	Motorola	1
	2N6039	Power Transistor	Motorola	1
	TL612U	120vF Capacitor	Mallory	1
	RNC65H2262FS	22.6K ohm, 1%	Dale	2
	RNC65D7501F	7.5K ohm, 1%	Dale	2
	30D[illegible]1305	20vF Capacitor	Sprague	2
	RC07GF471J	470, 1/4W Resitor	Allen-Bradley	2
	TL6121U	120VF Capacitor	Mallory	1
	1N3208	Diode	Motorola	1
	LX0513X103K	.01VF Capacitor	Sprague	1



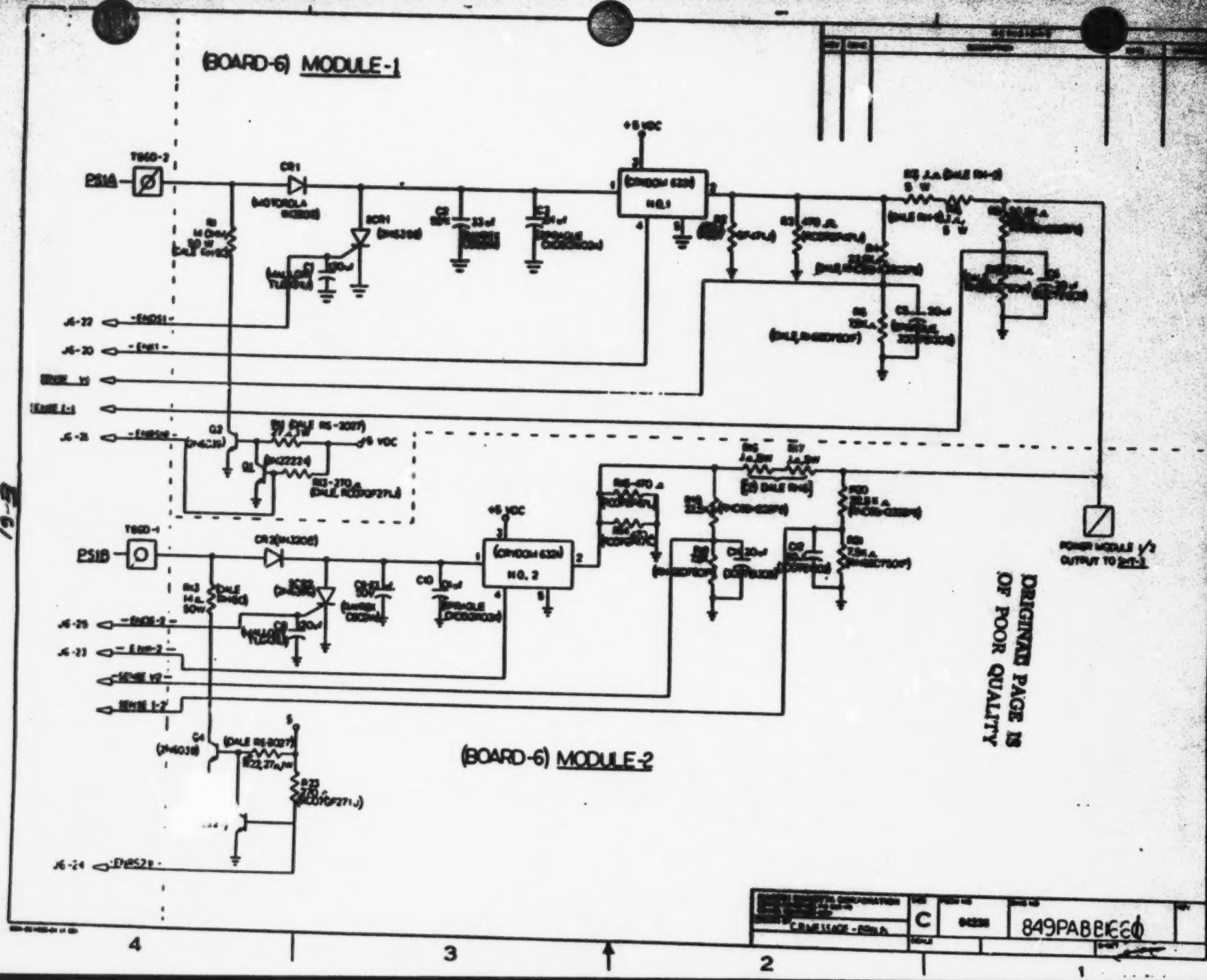
ORIGINAL PAGE IS OF POOR QUALITY


(BOARD-6) MODULE-9
MODULE 2 BATTERY OUTPUT
J6-6
J6-5
J6-4
SENSE V-9
SENSE I-9
J6-3
J6-2
J6-1
SENSE V-11
SENSE I-11
CRYSON NO.9
POTTER / BRUMFIELD NO.11
MODULE 2 COMBINED BUS OUTPUT
MODULE 2 REGULATOR INPUT
(BOARD-6) MODULE-11
849PAB81660


(BOARD-6) MODULE-5
MODULE 1 COMBINED BUS OUTPUT
+12VDC
+5VDC
NO.5
TO POWER MODULE NO.1 REGULATOR INPUT
SENSE V-5
SENSE I-5
MODULE 1 REGULATOR OUTPUT
POTTER & BRUMFIELD
NO.6
POWER MODULE NO.1 BUS OUTPUT
SENSE V-6
SENSE I-6
C
849PA8B1660


(BOARD-6) MODULE-3
(BOARD-6) MODULE-4
MODULE 1 BATTERY OUTPUT
(FROM SHT 2) POWER MODULE 1/2 OUTPUT
MODULE 1 COMBINED BUS OUTPUT (TO SHT 4)
+5 VDC
5 VDC
SENSE V-8
SENSE I-3
SENSE V-4
SENSE I-4
J6-12
J6-10
J6-11
J6-19
J6-17
J6-18
(CRYDOM 6321) NO. 3
(CRYDOM 6321) NO. 4
C
04236
34CPAB8660

E-60

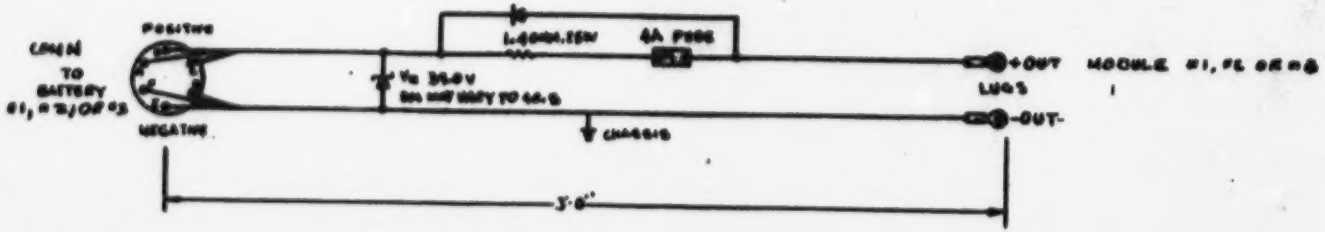


ORIGINAL PAGE IS OF POOR QUALITY

BATTERY SYSTEM ASSEMBLY

Drawing #PL849PABB1700

Item #	Part #	Description	MFG	QTY
1	P-10	10.5" X 19" Panel	Optima	1
2	DRS-30	Drawer Slides	Optima	1
3		T4 Alluminum-3/16"	MMC Stock	1
4	H-10	Set of Panel Handles	Optima	1
5	25CELL1414-0	Battery Modules	Eagle-Pitcher	3
6	KPSE06B-10-6S	Connector Blackshells	Amphenol	3
7	8502-002	20 Guage wire-red	Belden	1
8	8502-010	20 Guage wire-black	Belden	1
9	86-12-312	12 Volt Power Supply	Sola-Electric	1
10		Lugs		5
11		T4 Alumimun - 1/4"	MMC Stock	1
12		#8/32 Bolts	NL Fasteners	26
13		#8/32 Nuts	NL Fasteners	26
14		#8 Lockwashers	NL Fasteners	26
15		#8 Flatwashers	NL Fasteners	26
16	17614C	Power Cord	Belden	1



CHARGER RETROFIT

1) CONSTANT VOLTAGE POWER SUPPLIES (IN A CURRENT LIMIT MODE) WILL SUPPLY ACROSS THE LIMITING RESISTOR A CONSTANT CURRENT OF 1.0/3.6 COULOMB/FAST CHARGE RATE) AS LONG AS THE BATTERY VOLTAGE REMAINS AT OR BELOW 32.4/37.8 VOLTS. ABOVE THESE BATTERY VOLTAGES, A CONSTANT VOLTAGE OF 33.0 VOLTS WILL BE APPLIED TO THE BATTERY.

2) THE 30 AMP-HOUR BATTERY WILL CHARGE IN APPROXIMATELY 07.0/3.50 HOURS.

3) BATTERIES WILL BE DISCONNECTED FROM THE CHARGE SOURCE WHEN THE CHARGING CURRENT FALLS BELOW .1A (TRACE MINIMUM RESOLUTION OF THE CURRENT SENSORS).

BATTERY MODULE CABLING

1) NOMINAL BATTERY VOLTAGE = 25 CELLS X 1.35 VOLTS/CELL = 33.75v

2) $\frac{\text{NOMINAL BATTERY VOLTAGE - VOLTAGE AT REGULAR SENSE}}{\text{DESIRED BATTERY OUTPUT}} = \frac{\text{APPROXIMATE VALUE OF LIMIT RESISTOR}}{-} = \frac{33.75 - 32.75}{.5\ \text{AMP}} = 2\ \Omega$

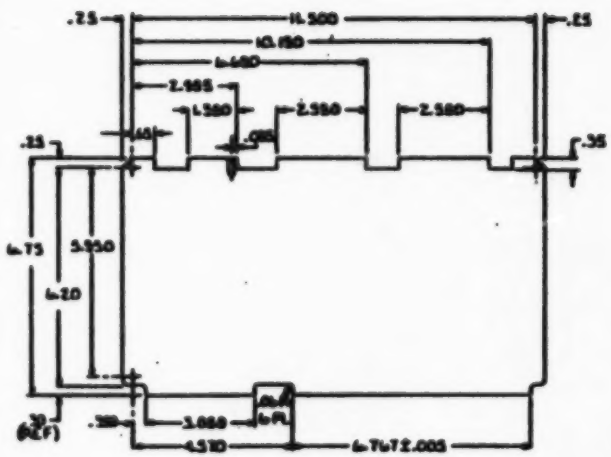
3) $\frac{\text{NOMINAL BATTERY VOLTAGE}}{\text{ACTUAL LIMIT RESISTOR}} = \frac{\text{SHORT CIRCUIT CURRENT}}{-} = \frac{33.75}{1.4} = 23.0\ \text{A}$

4) NEXTERE CAPACITY = 4.5 AMPERES ∴ FUSE AT 5 AMPS

5) P_R IN LIMIT RESISTOR = $(16)^2$ 1.4 = 22.4 WATTS

SIZE	FSCM NO.	DRAWING NO.	REV.
C	04236		
SCALE		SHEET	

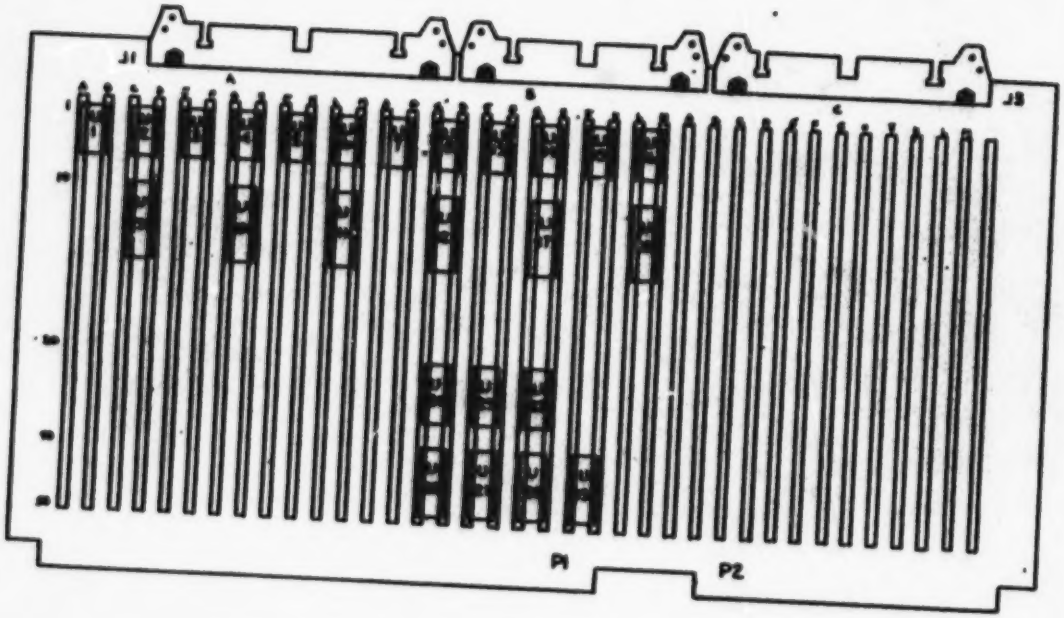
DESIGNATION	JUMPER TYPE	CONNECTION POINTS
W1	SOLDERED	E-H
52	WIRE-WRAP	1-4
87	WIRE-WRAP	87-86
27	WIRE-WRAP	27-28
109	WIRE-WRAP	109-103
30	WIRE-WRAP	30-29
40	WIRE-WRAP	40-41
42	WIRE-WRAP	42-43
44	WIRE-WRAP	44-45
46	WIRE-WRAP	46-47
48	WIRE-WRAP	48-49
50	WIRE-WRAP	50-51
53	WIRE-WRAP	53-54
51	WIRE-WRAP	1-4
84	WIRE-WRAP	84-85
79	WIRE-WRAP	79-81
18	WIRE-WRAP	18-16
21	WIRE-WRAP	21-19



SBC 517 SIO PC BOARD ASSEMBLY

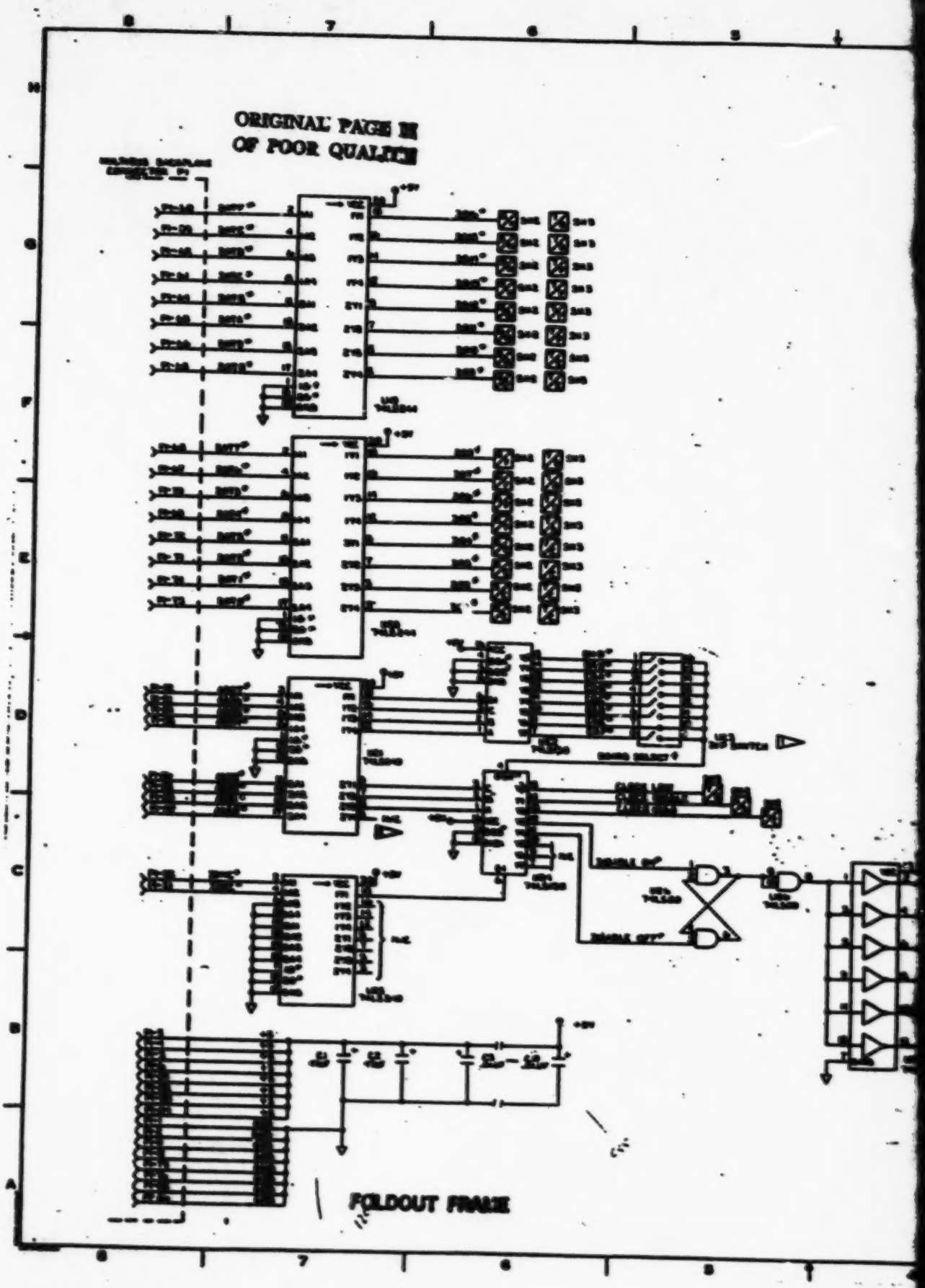
	C	[illegible]	849PAB81470

E-64



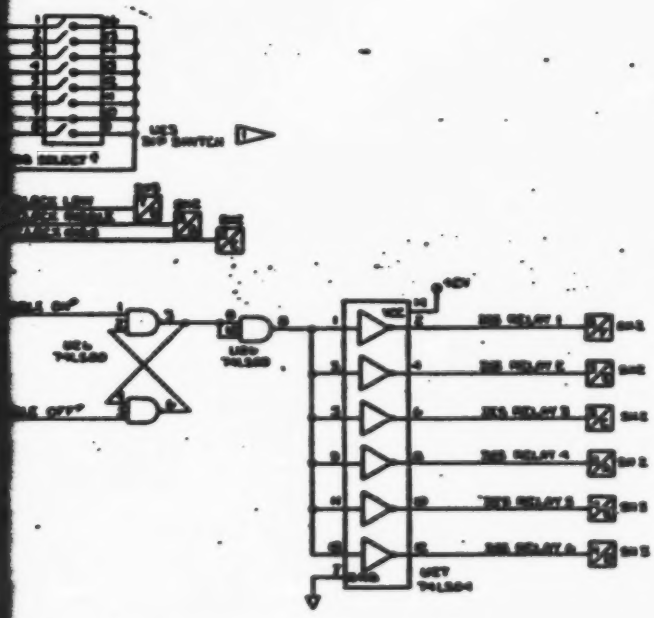
E-65

ORIGINAL PAGE IS
OF POOR QUALITY



FOLDOUT FRAME

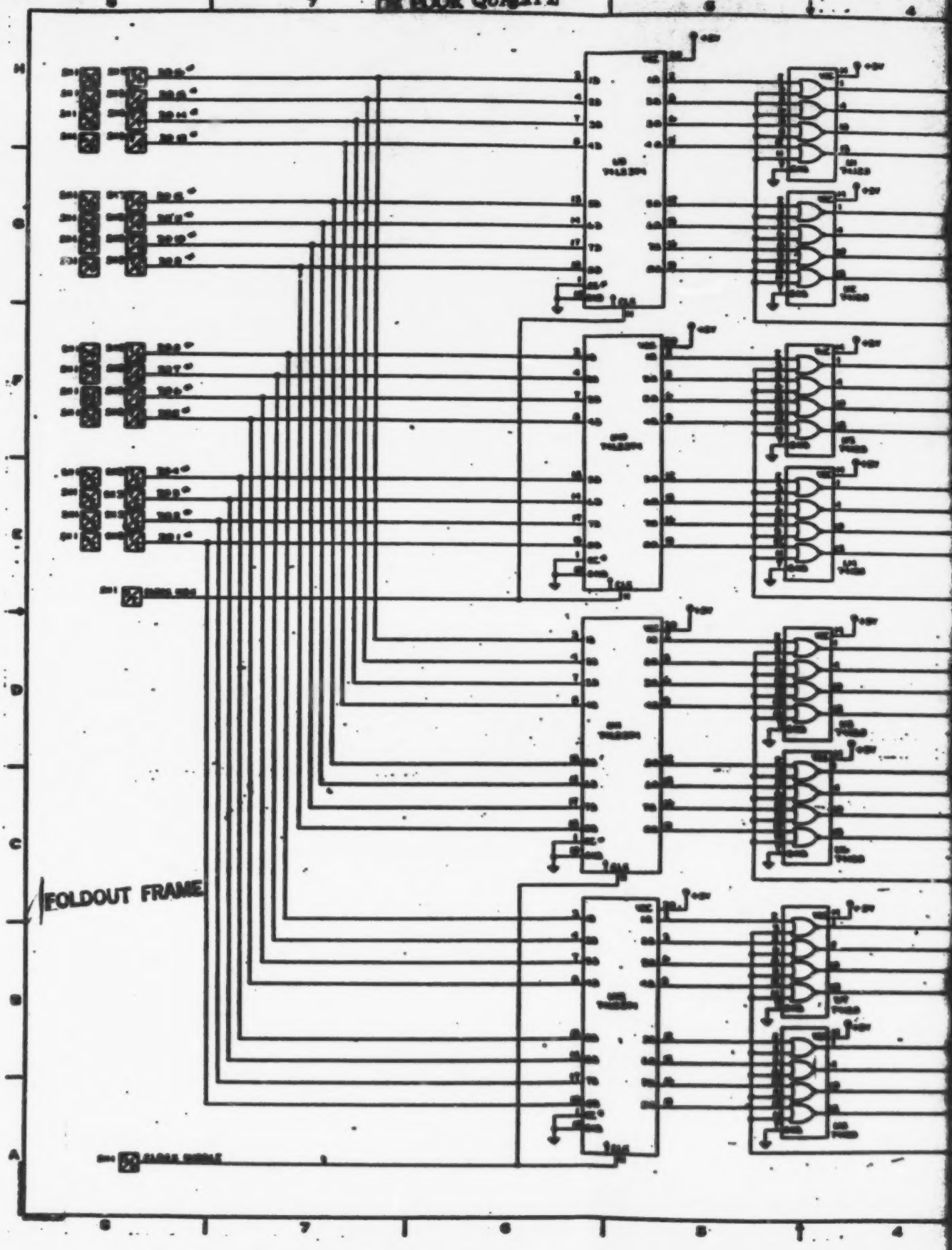
ORIGINAL PAGE IS
OF POOR QUALITY



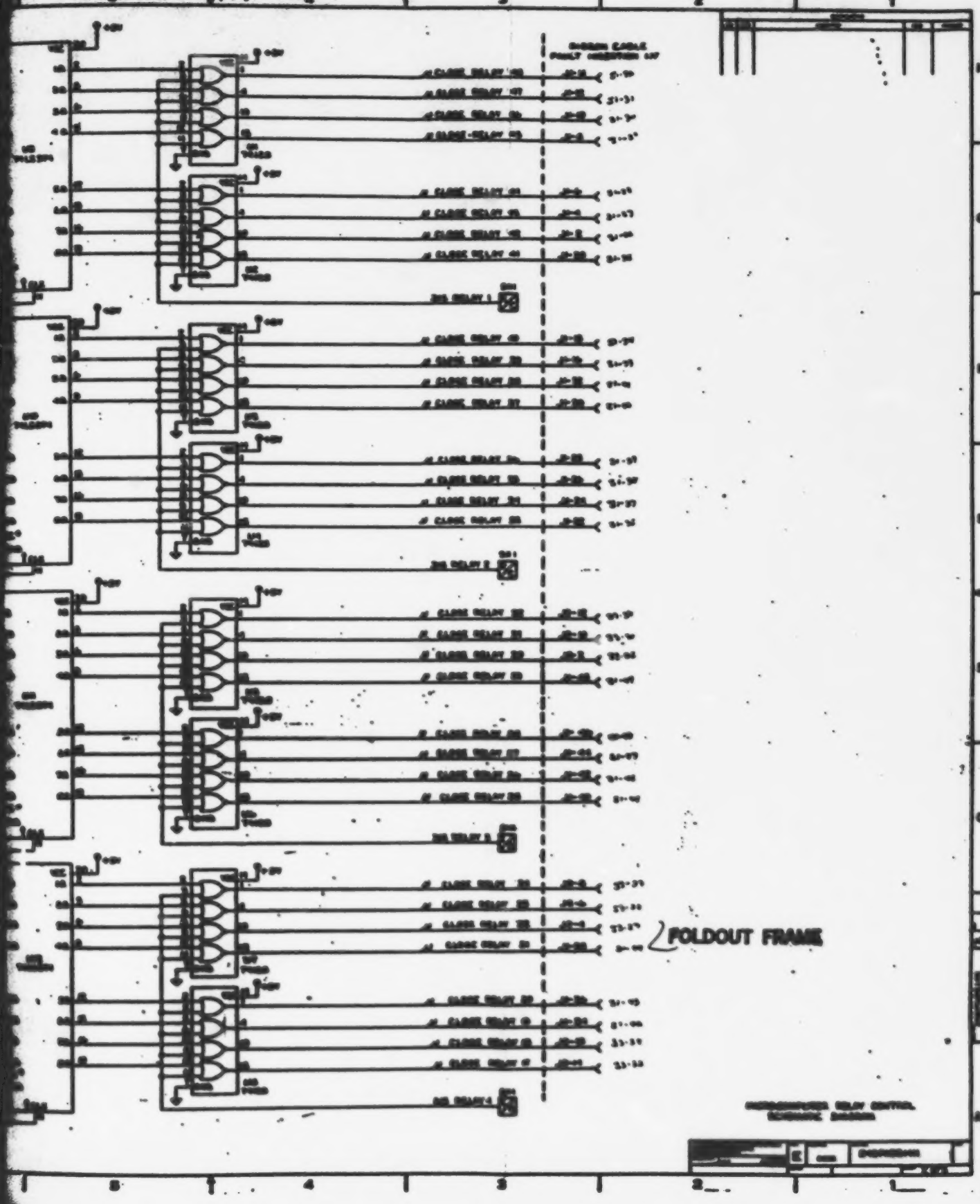
FOLDOUT FRAME

MICROCOMPUTER RELAY CONTROL
SCHEMATIC DIAGRAM

ORIGINAL PAGE IS
OF POOR QUALITY

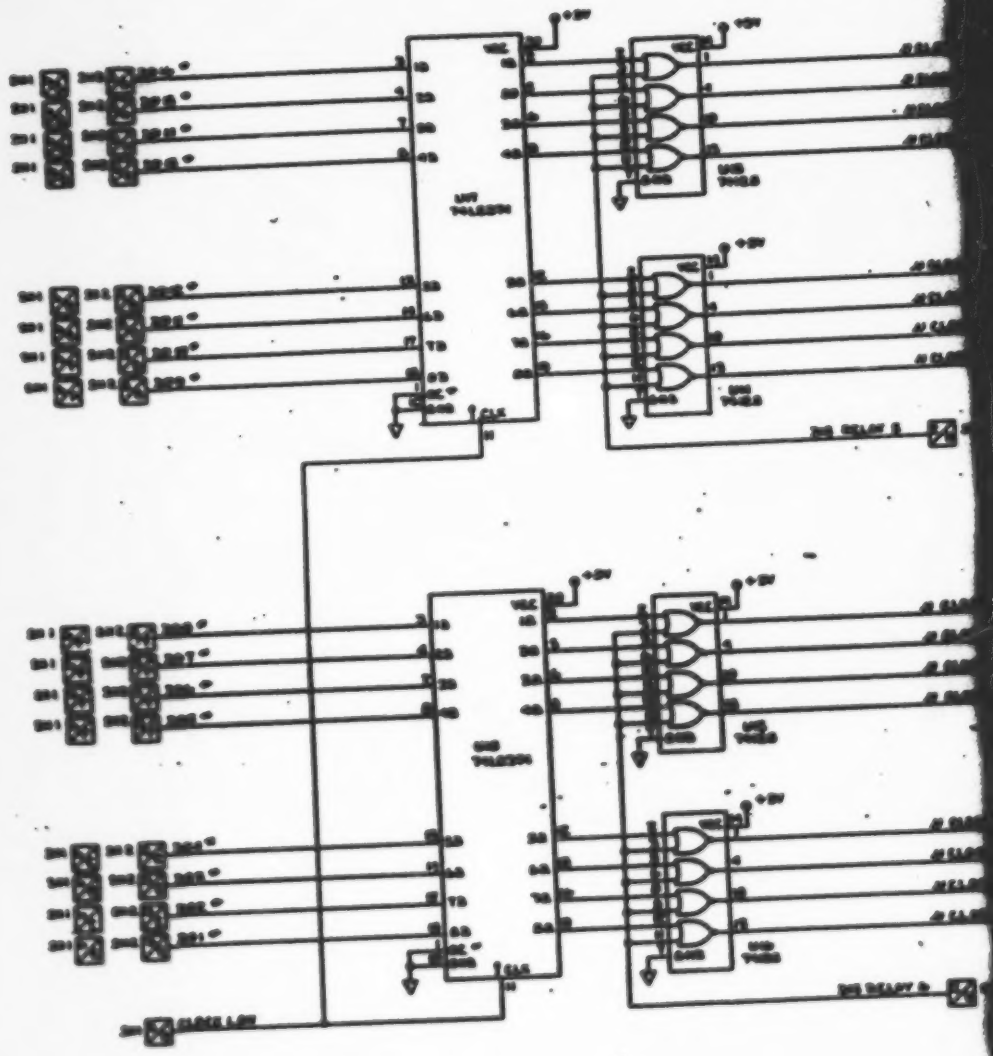


E-67



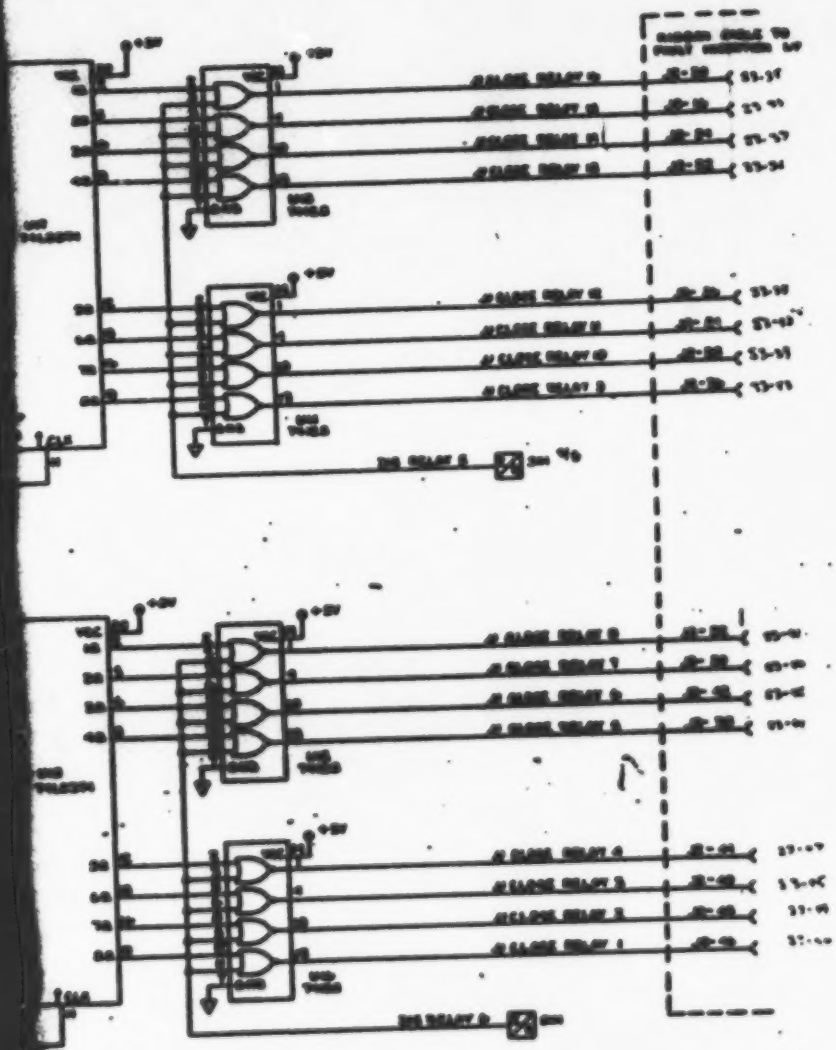
FOLDOUT FRAME

ORIGINAL PAGE IS
OF POOR QUALITY



FOLDOUT FRAME

E-68



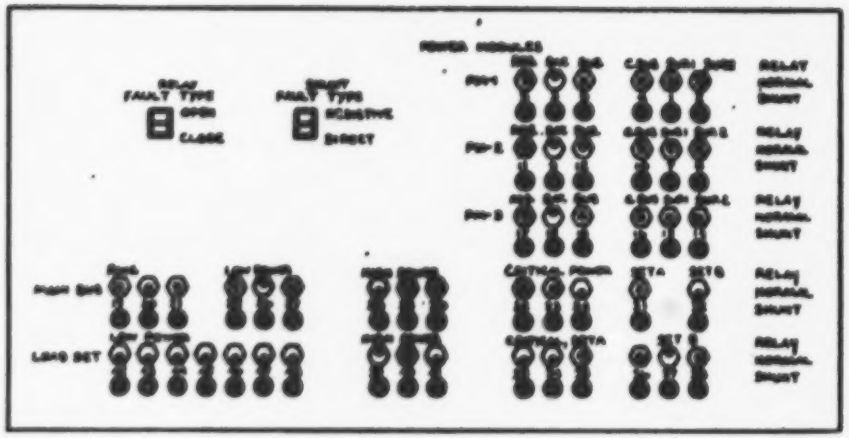
FOLDOUT FRAME

MICROCOMPUTER RELAY CONTROL
SCHEMATIC DIAGRAM

FAULT INSERTION ASSEMBLY

Drawing #PL849PABB1300

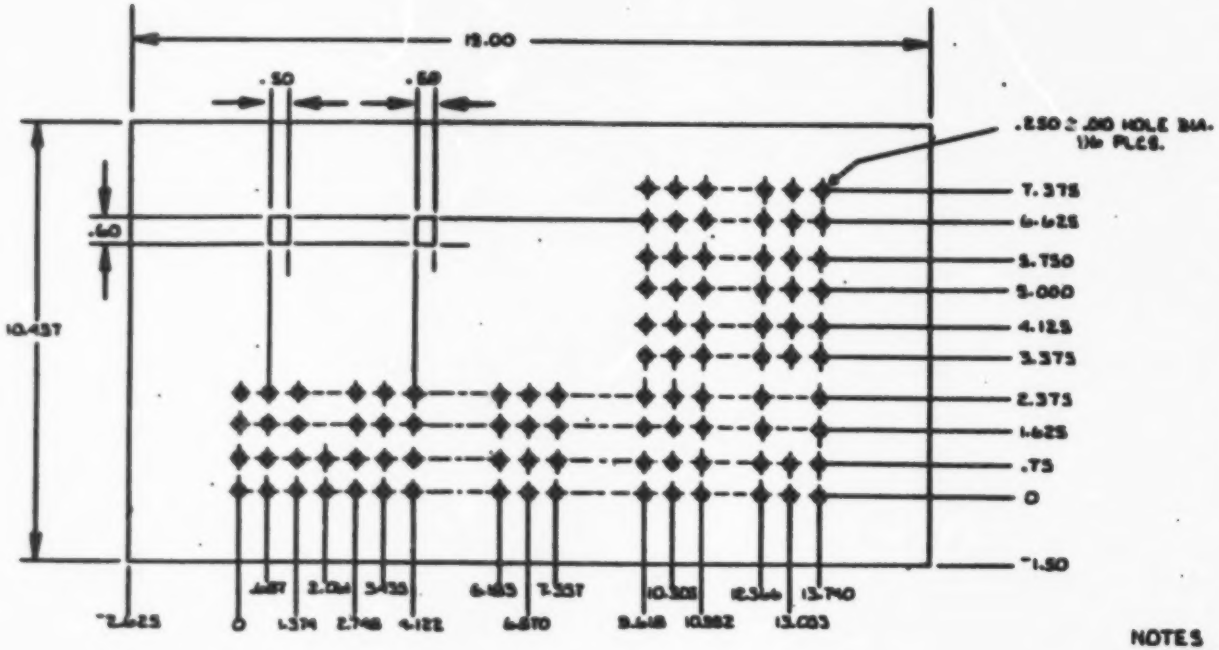
Item #	Part #	Description	MFG	QTY
1	P-10	10.5"x19" Panel	Optima	1
2	DRS-30	Drawer Slides	Optima	1
3		T4 Alumunimn 3/16"	MMC stock	1
4	H-10	Set of Panel Handles	Optima	1
5	HGMP-0280	Integrated Lamps	Hewlett Packard	48
6	MST-105E	3 Position toggle switches	Leecraft	48
7	572-1121-0804-040	2 Position paddle switches	Dialight	2
8	39TB3	Terminal Barrier strip	Control Design	1
9	9L28050	Wiring Harness	3M/Newark	1
10		Perforated Base	MMC stock	1
11	849PABB1310	Wire Wrap Assemblies	MMC	2
12	13P12RC	12 Gauge wire - red	Carol	1
13		Terminal Lugs		8
14	TH-112-48L	Lettering set	Transfertech	3
15		#8/32 Bolts	NL Fasteners	20
16		#8/32 Nuts	NL Fasteners	20
17		#8 Lockwashers	NL Fasteners	20
18		#8 Flatwashers	NL Fasteners	20
19	13R12BC	12 Guage Wire - black	Carol	1
20	TCG121V	120 VF, 25V Capacitor	Mallory	1



POWER SUBSYSTEM AUTOMATION STUDY
FAULT INSERTION PANEL –
ASSEMBLY DRAWING AND DRILL DETAIL

	C	[illegible]	849PA881300
25FEB85	NONE		1 OF 3

E-90



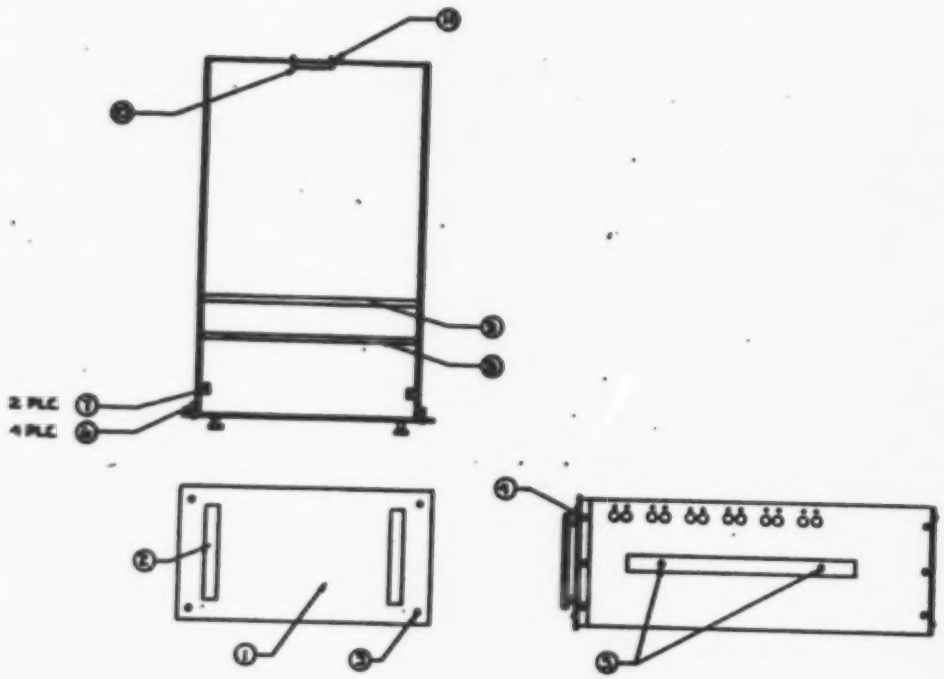
ORIGINAL PAGE IS OF POOR QUALITY

NOTES

1. PANEL MATERIAL IS .125 ALUMINUM T4. AFTER DRILL PANEL IS PAINTED GRAY #45 AND KRYLON COATED.

POWER SUBSYSTEM AUTOMATION STUDY
FAULT INSERTION PANEL -
ASSEMBLY DRAWING AND DRILL DETAIL

	C		845M881500
20 FEB 85	NONE		SHEET 2 OF 3



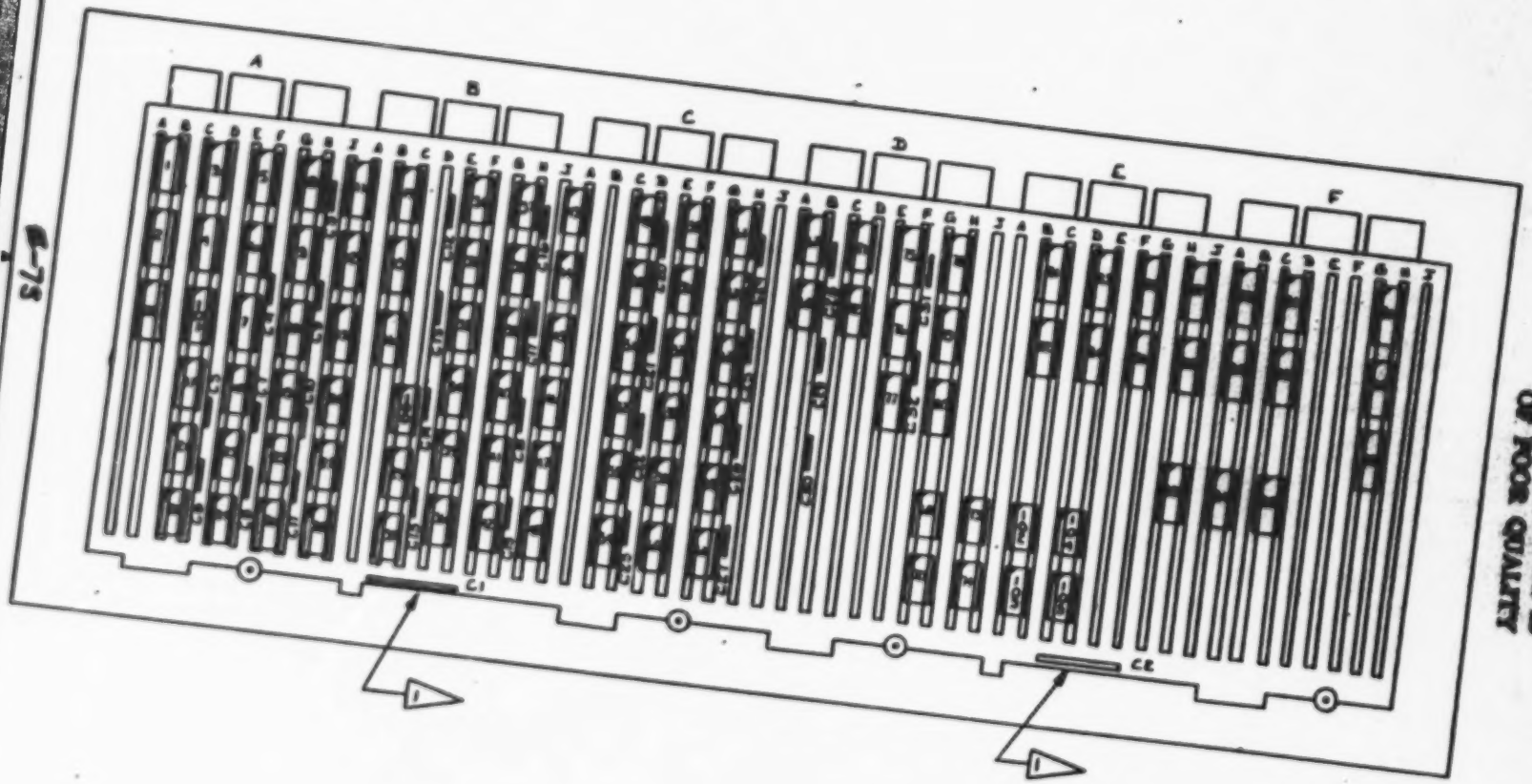
POWER SUBSYSTEM AUTOMATION STUDY –
FAULT INSERTION MODULE ASSEMBLY

C

E-72

NOTES:

1. C1 AND C2 ARE MOUNTED ON THE REAR (WIRING) SIDE OF THE BOARD.



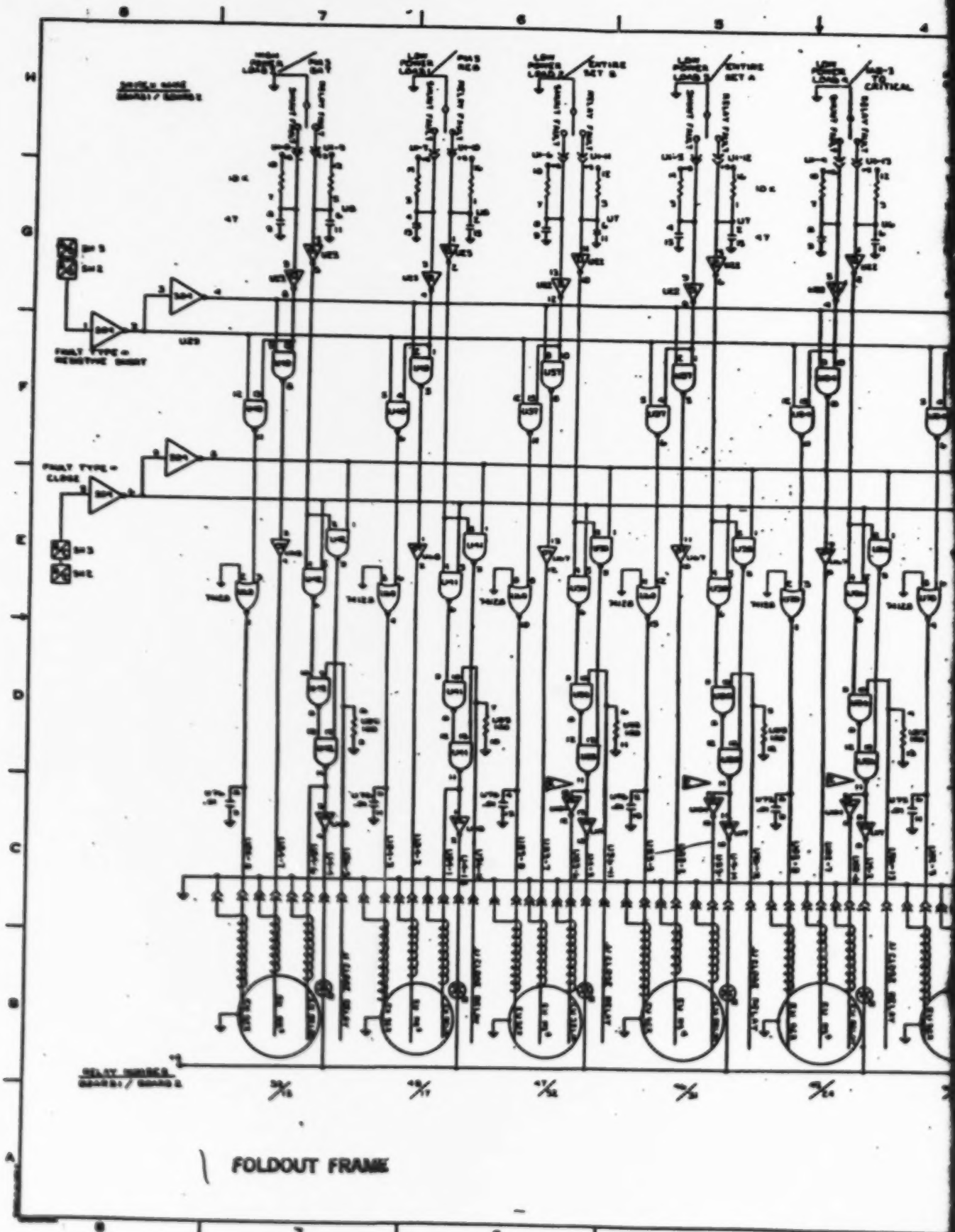
ORIGINAL PAGE IS OF POOR QUALITY

FAULT INSERTION PANEL INTERFACE

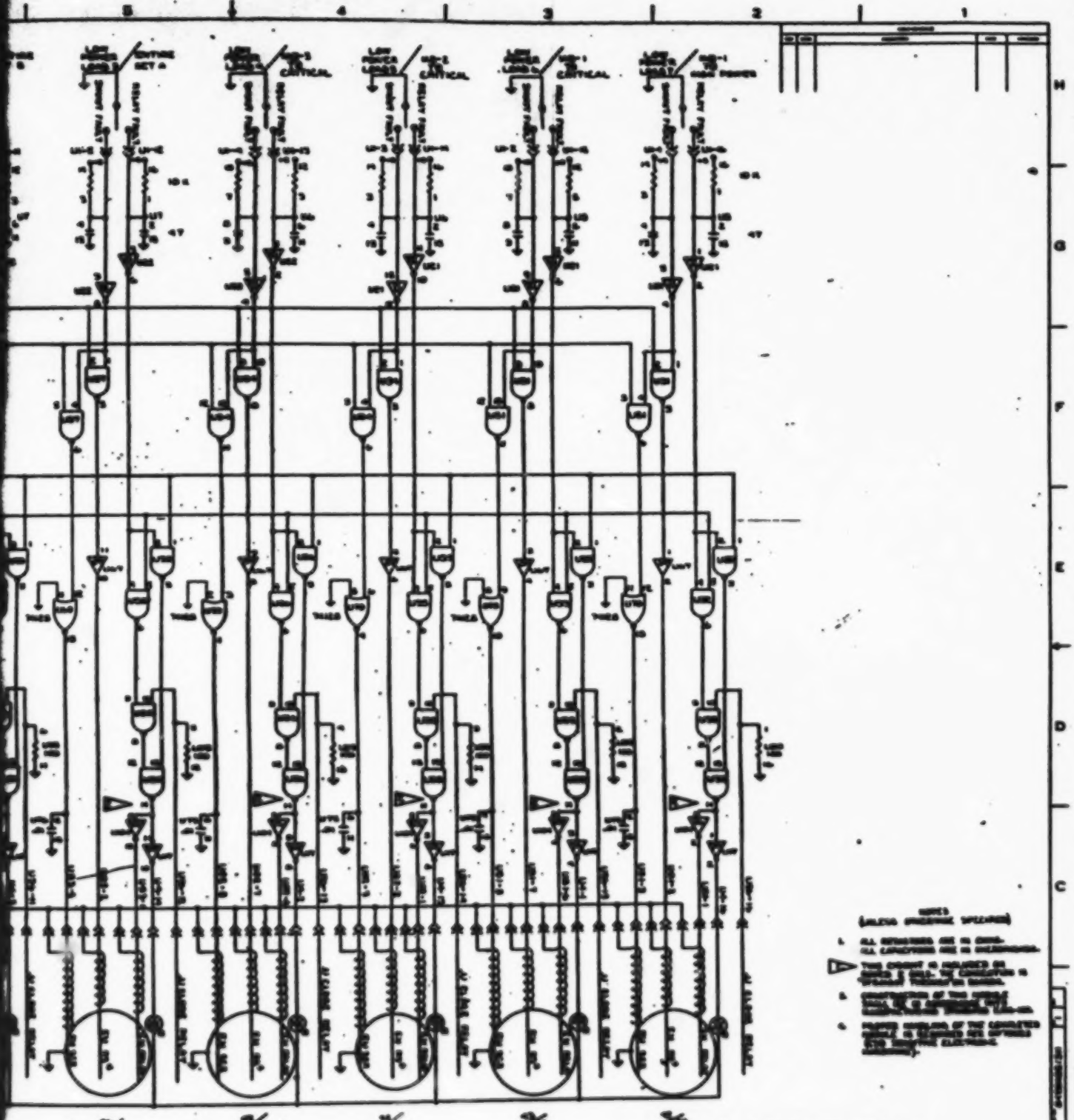
MARTIN MARIETTA CORPORATION	C	849PA881310	
26FEB88		1 OF 1	

4 3 2 1

PAGE III
QUALITY



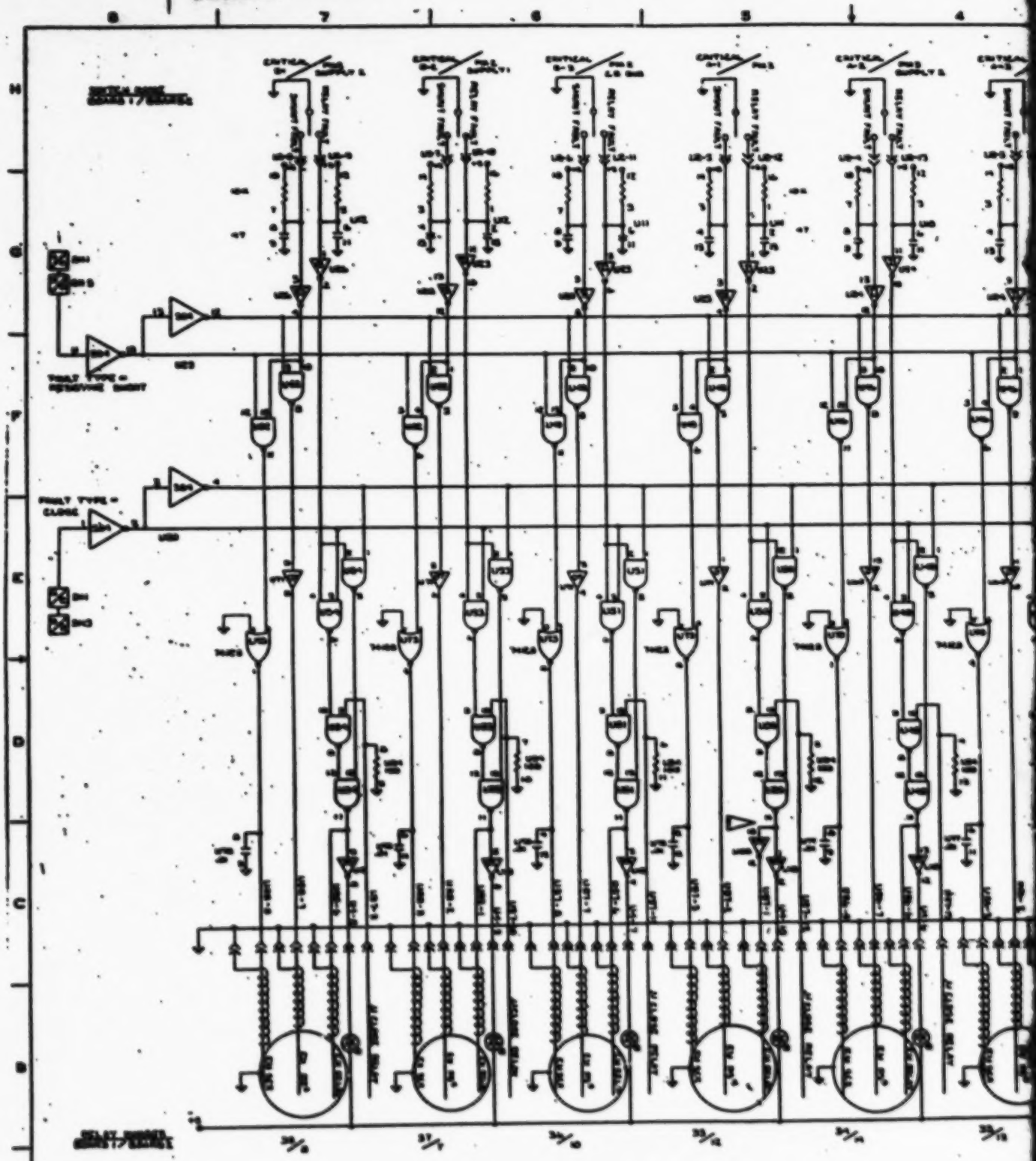
FOLDOUT FRAME



2 FOLDOUT FRAME

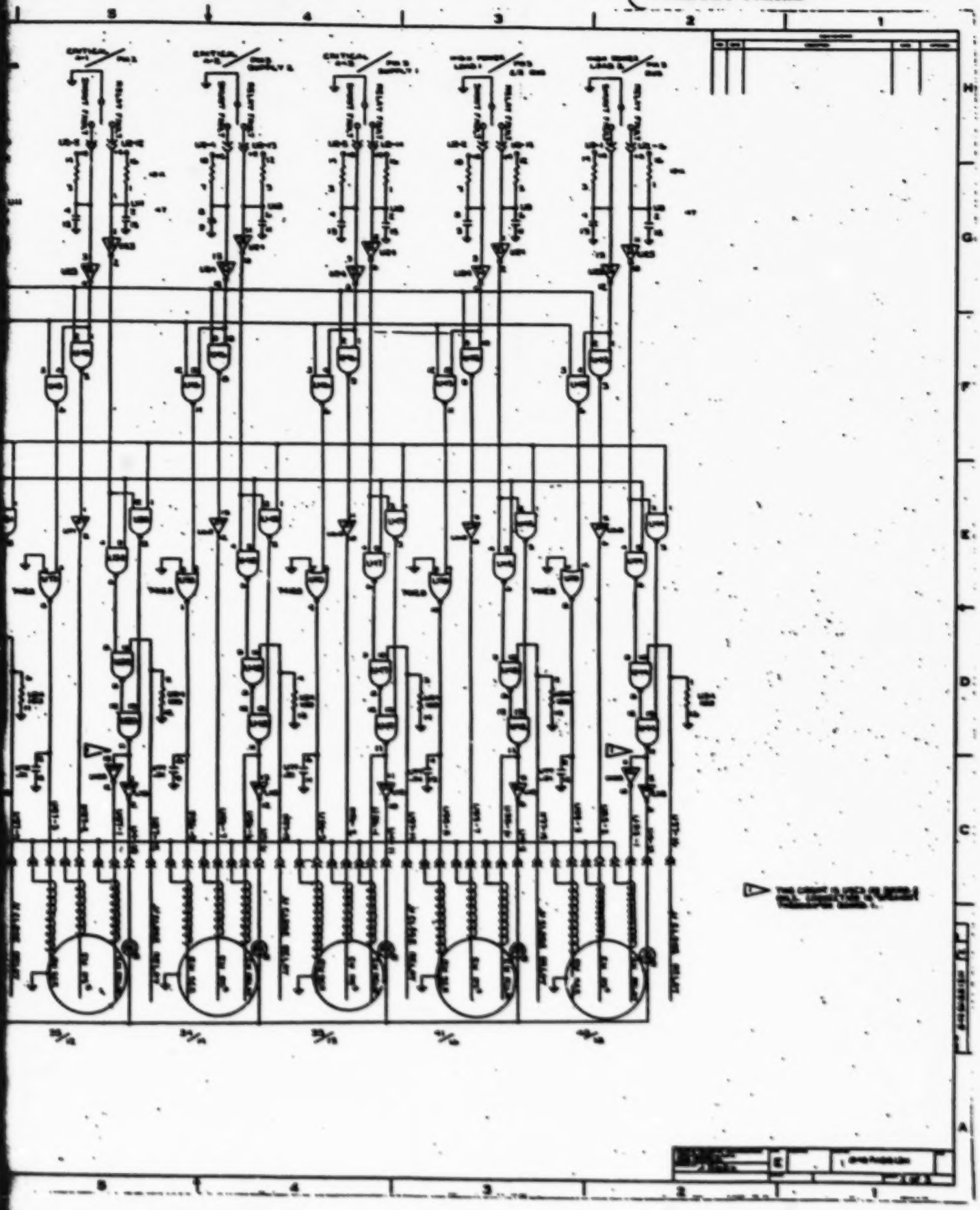
FAULT INJECTION I/F
SCHEMATIC DIAGRAM

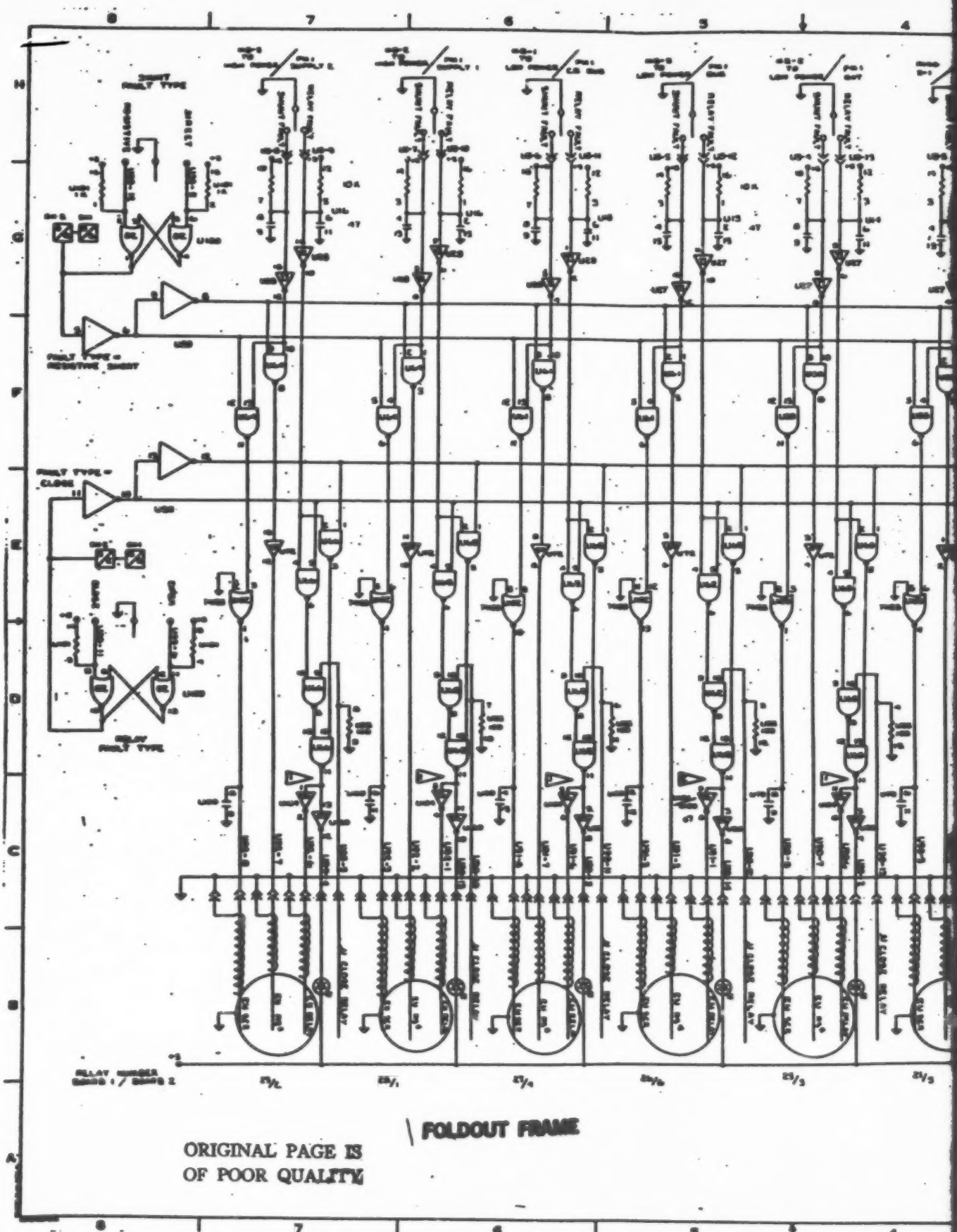
FOLDOUT FRAME



ORIGINAL PAGE IS
OF POOR QUALITY

FOLDOUT FRAME

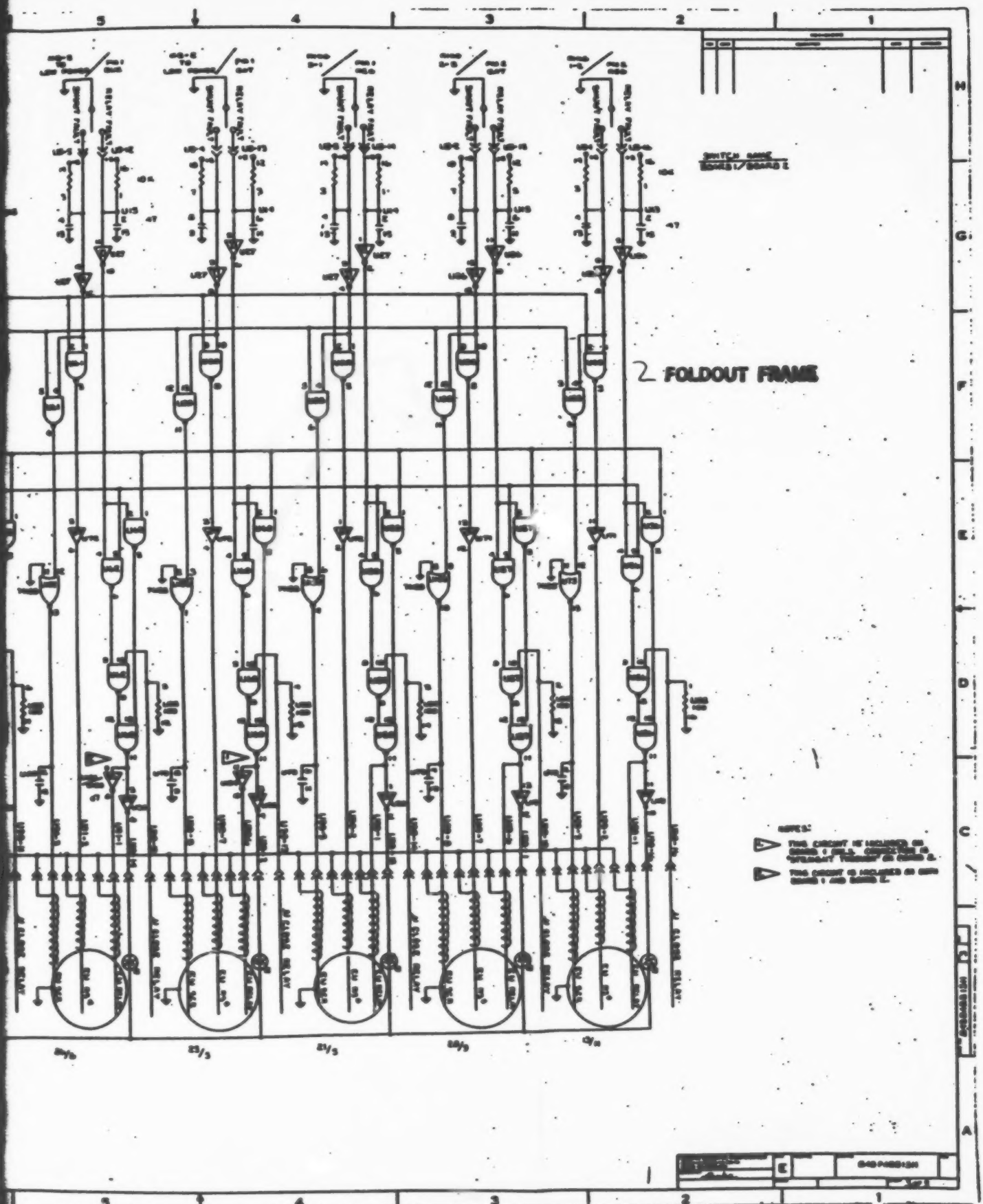




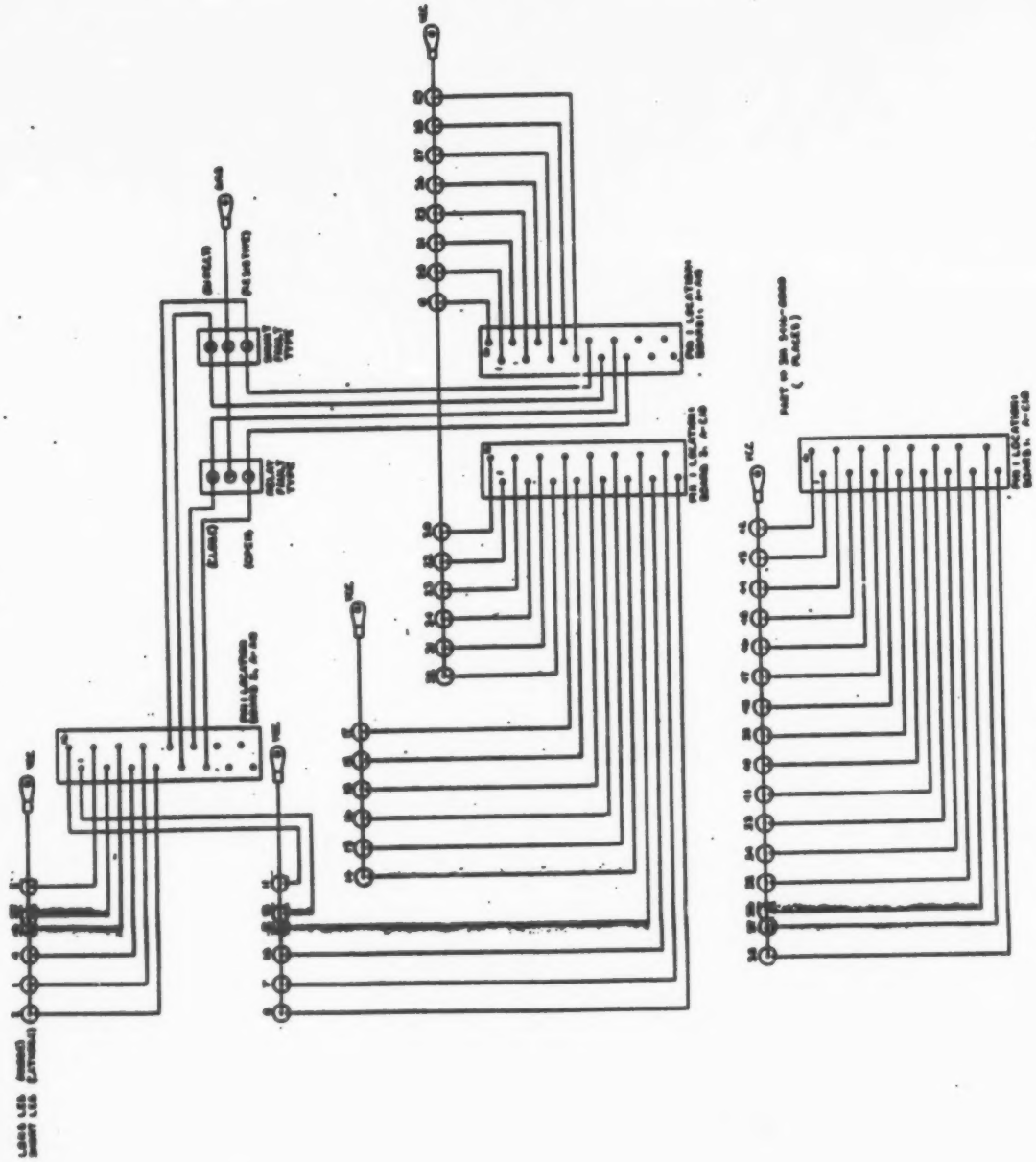
FOLDOUT FRAME

ORIGINAL PAGE IS
OF POOR QUALITY

E-76


2 FOLDOUT FRAME

ORIGINAL PAGE IS
OF POOR QUALITY



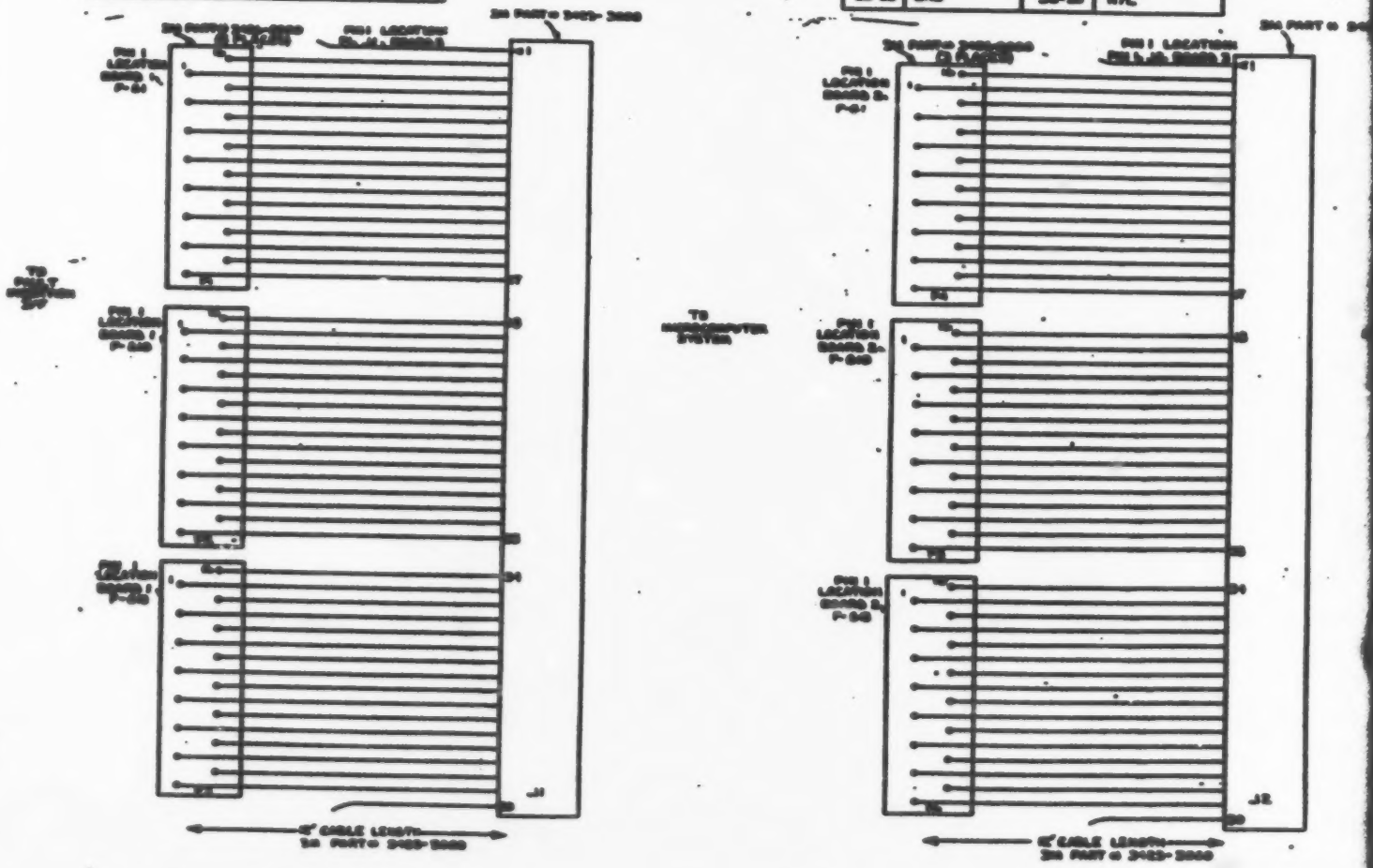
ORIGINAL PAGE IS
OF POOR QUALITY

TABLE J1
SIGNAL DESCRIPTION

J2-1	N/C	J2-26	CLOSE RELAY 32
J2-2	CLOSE RELAY 48	J2-27	GND
J2-3	GND	J2-28	CLOSE RELAY 24
J2-4	CLOSE RELAY 40	J2-29	GND
J2-5	GND	J2-30	CLOSE RELAY 31
J2-6	CLOSE RELAY 44	J2-31	GND
J2-7	GND	J2-32	CLOSE RELAY 30
J2-8	CLOSE RELAY 46	J2-33	GND
J2-9	GND	J2-34	CLOSE RELAY 6
J2-10	CLOSE RELAY 42	J2-35	GND
J2-11	GND	J2-36	CLOSE RELAY 30
J2-12	CLOSE RELAY 47	J2-37	GND
J2-13	GND	J2-38	CLOSE RELAY 21
J2-14	CLOSE RELAY 43	J2-39	GND
J2-15	GND	J2-40	CLOSE RELAY 15
J2-16	CLOSE RELAY 39	J2-41	GND
J2-17	GND	J2-42	CLOSE RELAY 14
J2-18	CLOSE RELAY 45	J2-43	GND
J2-19	GND	J2-44	CLOSE RELAY 13
J2-20	CLOSE RELAY 41	J2-45	GND
J2-21	GND	J2-46	CLOSE RELAY 10
J2-22	CLOSE RELAY 37	J2-47	GND
J2-23	GND	J2-48	CLOSE RELAY 12
J2-24	CLOSE RELAY 34	J2-49	GND
J2-25	GND	J2-50	N/C

TABLE J2
SIGNAL DESCRIPTION

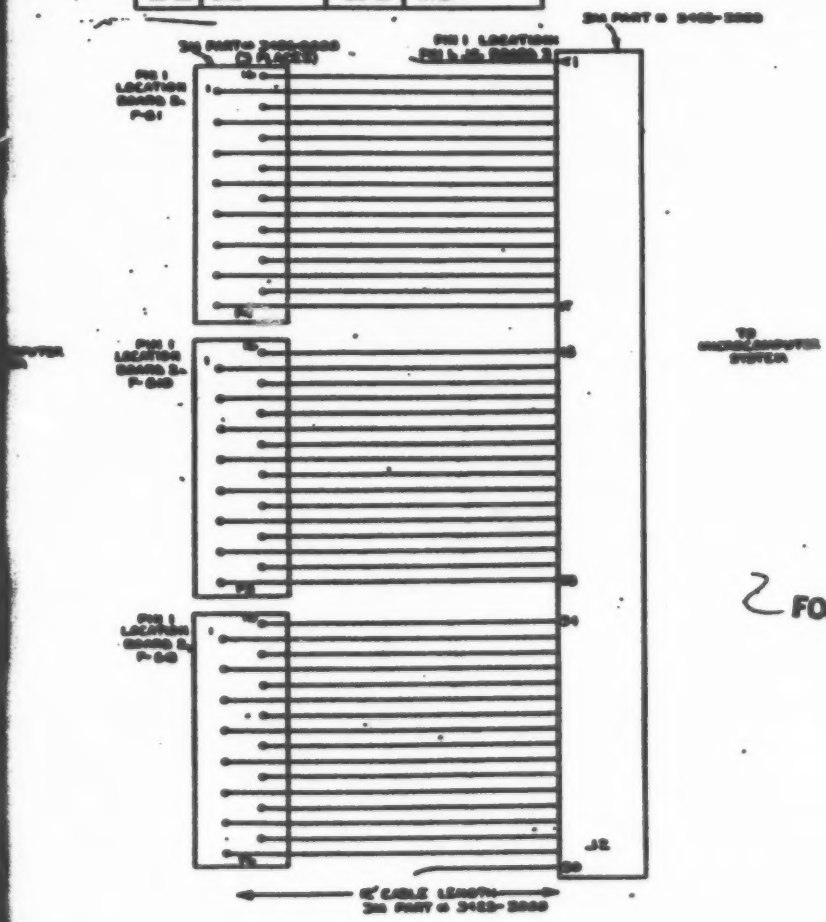
J2-1	N/C	J2-26	CLOSE RELAY 8
J2-2	CLOSE RELAY 26	J2-27	GND
J2-3	GND	J2-28	CLOSE RELAY 10
J2-4	CLOSE RELAY 22	J2-29	GND
J2-5	GND	J2-30	CLOSE RELAY 7
J2-6	CLOSE RELAY 23	J2-31	GND
J2-7	GND	J2-32	CLOSE RELAY 6
J2-8	CLOSE RELAY 24	J2-33	GND
J2-9	GND	J2-34	CLOSE RELAY 4
J2-10	CLOSE RELAY 20	J2-35	GND
J2-11	GND	J2-36	CLOSE RELAY 9
J2-12	CLOSE RELAY 18	J2-37	GND
J2-13	GND	J2-38	CLOSE RELAY 3
J2-14	CLOSE RELAY 17	J2-39	GND
J2-15	GND	J2-40	CLOSE RELAY 2
J2-16	CLOSE RELAY 13	J2-41	GND
J2-17	GND	J2-42	CLOSE RELAY 5
J2-18	CLOSE RELAY [illegible]	J2-43	GND
J2-19	GND	J2-44	CLOSE RELAY 4
J2-20	CLOSE RELAY 9	J2-45	GND
J2-21	GND	J2-46	CLOSE RELAY 1
J2-22	CLOSE RELAY 6	J2-47	GND
J2-23	GND	J2-48	CLOSE RELAY 2
J2-24	CLOSE RELAY 10	J2-49	GND
J2-25	GND	J2-50	N/C



FOLDOUT FRAME

TABLE J2
SIGNAL DESCRIPTION

J2-1	N/C	J2-26	CLOSE RELAY 12
J2-2	CLOSE RELAY 20	J2-27	GND
J2-3	GND	J2-28	CLOSE RELAY 10
J2-4	CLOSE RELAY 22	J2-29	GND
J2-5	GND	J2-30	CLOSE RELAY 7
J2-6	CLOSE RELAY 23	J2-31	GND
J2-7	GND	J2-32	CLOSE RELAY 8
J2-8	CLOSE RELAY 24	J2-33	GND
J2-9	GND	J2-34	CLOSE RELAY 6
J2-10	CLOSE RELAY 26	J2-35	GND
J2-11	GND	J2-36	CLOSE RELAY 5
J2-12	CLOSE RELAY 28	J2-37	GND
J2-13	GND	J2-38	CLOSE RELAY 3
J2-14	CLOSE RELAY 27	J2-39	GND
J2-15	GND	J2-40	CLOSE RELAY 2
J2-16	CLOSE RELAY 29	J2-41	GND
J2-17	GND	J2-42	CLOSE RELAY 0
J2-18	CLOSE RELAY 30	J2-43	GND
J2-19	GND	J2-44	CLOSE RELAY 4
J2-20	CLOSE RELAY 31	J2-45	GND
J2-21	GND	J2-46	CLOSE RELAY 1
J2-22	CLOSE RELAY 32	J2-47	GND
J2-23	GND	J2-48	CLOSE RELAY 2
J2-24	CLOSE RELAY 14	J2-49	GND
J2-25	GND	J2-50	N/C



FOLDOUT FRAME

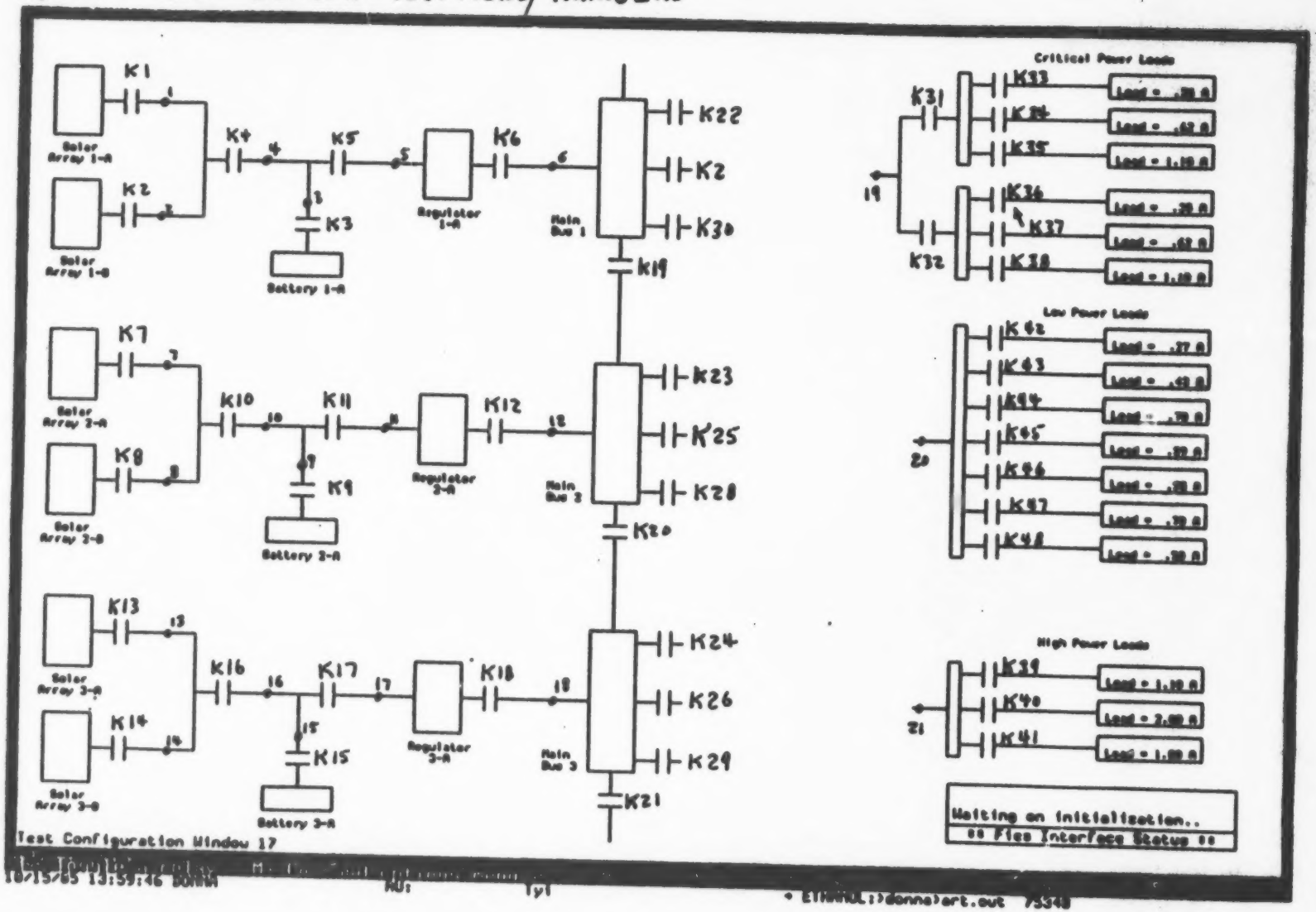
RETAIL CABLE DIAGRAM

MAP ASSEMBLY

Drawing #PL849PABB1200

Item #	Part #	Description	MFG	QTY
1	P10-White	10.5" X 19" Panel	Optima	1

RELAY AND SENSOR POSITIONS/NUMBERS



VENTILLATION GRILLE

Item #	Part #	Description	MFG	QTY
1	G-319	10.5" X 19" Panel	Optima	5

VOLTAGE REGULATOR ASSEMBLY

Drawing # PL849PABB1100

Item #	Part #	Description	MFG	QTY
1	KD1-3.0	3AMP Circuit Breaker	Heinemann	3
2	SU2A1	FAN, 120 VAC	Sprite	1
3	17614C	Power Cord	Belden	1
4	849PABB1110	Voltage Regulator	MMC	1
5	849PABB1120	Battery Protection	MMC	1
6	849PABB1130	Supply Disconnection	MMC	1
7	P-10	10.5" X 19" Panel	Optima	1
8		#8/32 Bolts		20
9		#8/32 Nuts		20
10		#8 Lockwashers		20
11		#8 Flatwashers		20

BOARD - Ø - VOLTAGE

PIN	SINGLE-ENDED	DIFFERENTIAL	PIN	SINGLE-ENDED	DIFFERENTIAL		MEASURED AT RELAY #
1	Not Used	Not Used	2	Not Used	Not Used	SENSOR	
3	Analog Return	Analog Return	4	CH 0	CH 0 HI	S4	
5	↕	↕	6	CH 8	CH 0 LO	'GND'	K4
7			8	CH 1	CH 1 HI	S1	
9			10	CH 9	CH 1 LO	'GND'	K1
11			12	CH 2	CH 2 HI	S2	
13			14	CH 10	CH 2 LO	'GND'	K2
15			16	CH 3	CH 3 HI	S5	
17			18	CH 11	CH 3 LO	'GND'	K5
19			20	CH 4	CH 4 HI	S3	
21			22	CH 12	CH 4 LO	'GND'	K3
23			24	CH 5	CH 5 HI	S6	
25			26	CH 13	CH 5 LO	'GND'	K6
27			28	CH 6	CH 6 HI	S11	
29			30	CH 14	CH 6 LO	'GND'	K11
31			32	CH 7	CH 7 HI	S9	
33			34	CH 15	CH 7 LO	'GND'	K9
35	Analog Return	Analog Return	36	Not Used	Not Used		
37	Not Used	Not Used	38	Not Used	Not Used		
39	Digital Common	Digital Common	40	Clock Out	Clock Out		
41	↕	↕	42	Ext. Trigger In	Ext. Trigger In		
43			44	BOC Status Out	BOC Status Out		
45	Digital Common	Digital Common	46	EOS Status Out	EOS Status Out		
47	Analog Return	Analog Return	48	Analog Return	Analog Return		
49	−15V	−15V	50	+15V	+15V		

NOTE:

'GND' IS THE LOCAL GND AT THE SENSOR

All odd-numbered pins (1, 3, . . . 49) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side with the board extractors at the top.

BOARD - Ø - VOLTAGE

Table 2.14 Analog Input Connector J2 Pin Assignments

PIN	SINGLE-ENDED	DIFFERENTIAL	PIN	SINGLE-ENDED	DIFFERENTIAL		MEASURED AT RELAY #
1	Not Used	Not Used	2	Not Used	Not Used	SENSOR	
3	Analog Return	Analog Return	4	CH 16	CH 8 HI	S10	
5	↕	↕	6	CH 24	CH 8 LO	'GND'	K10
7			8	CH 17	CH 9 HI	S7	
9			10	CH 25	CH 9 LO	'GND'	K7
11			12	CH 18	CH 10 HI	S8	
13			14	CH 26	CH 10 LO	'GND'	K8
15			16	CH 19	CH 11 HI	S13	
17			18	CH 27	CH 11 LO	'GND'	K13
19			20	CH 20	CH 12 HI	S14	
21			22	CH 28	CH 12 LO	'GND'	K14
23			24	CH 21	CH 13 HI	S12	
25			26	CH 29	CH 13 LO	'GND'	K12
27			28	CH 22	CH 14 HI	S18	
29			30	CH 30	CH 14 LO	'GND'	K18
31			32	CH 23	CH 15 HI	S16	
33			34	CH 31	CH 15 LO	'GND'	K16
35	Analog Return	Analog Return	36	Not Used	Not Used		
37	Not Used	Not Used	38	↕	↕		
39	↕	↕	40				
41			42				
43			44				
45	Not Used	Not Used	46	Not Used	Not Used		
47	Analog Return	Analog Return	48	Analog Return	Not Used		
49	−15V	−15V	50	+15V	+15V		

BØ

NOTE:

All odd-numbered pins (1, 3, . . . 49) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side with the board extractors at the top.

Table 2.15 Analog Input Connector J3 Pin Assignments

ORIGINAL PAGE IS
OF POOR QUALITY

FOLDOUT FRAME

BOARD 1 - VOLTAGE

PIN	SINGLE ENDED	DIFFERENTIAL	PIN	SINGLE ENDED	DIFFERENTIAL	SENSOR	MEASURED AT RELAY #
1	Not Used	Not Used	2	Not Used	Not Used		
3	Analog Return	Analog Return	4	CH 0	CH 0 HI	S21	K39
5			6	CH 8	CH 0 LO	'GND'	
7			8	CH 1	CH 1 HI	S22	
9			10	CH 9	CH 1 LO	'GND'	BATTERY 1
11			12	CH 2	CH 2 HI	S23	
13			14	CH 10	CH 2 LO	'GND'	BATTERY 2
15			16	CH 3	CH 3 HI	S24	
17			18	CH 11	CH 3 LO	'GND'	BATTERY 3
19			20	CH 4	CH 4 HI	S19	
21			22	CH 12	CH 4 LO	'GND'	K31
23			24	CH 5	CH 5 HI	S15	
25			26	CH 13	CH 5 LO	'GND'	K15
27			28	CH 6	CH 6 HI	S17	
29			30	CH 14	CH 6 LO	'GND'	K17
31			32	CH 7	CH 7 HI	S20	
33			34	CH 15	CH 7 LO	'GND'	BOARD 1
35	Analog Return	Analog Return	36	Not Used	Not Used		
37	Not Used	Not Used	38	Not Used	Not Used		
39	Digital Common	Digital Common	40	Clock Out	Clock Out		
41			42	Ext. Trigger In	Ext. Trigger In		
43			44	EOC Status Out	EOC Status Out		
45	Digital Common	Digital Common	46	EOS Status Out	EOS Status Out		
47	Analog Return	Analog Return	48	Analog Return	Analog Return		
49	−15V	−15V	50	+15V	+15V		

NOTE: 'GND' IS THE LOCAL GND AT THE SENSOR

All odd-numbered pins (1, 3, . . . 49) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side with the board extractors at the top.

NC = NO CONNECTION

Table 2.14 Analog Input Connector J2 Pin Assignments

PIN	SINGLE-ENDED	DIFFERENTIAL	PIN	SINGLE-ENDED	DIFFERENTIAL	SENSOR	MEASURED AT RELAY #
1	Not Used	Not Used	2	Not Used	Not Used		
3	Analog Return	Analog Return	4	CH 16	CH 8 HI		
5			6	CH 24	CH 8 LO	NC	
7			8	CH 17	CH 9 HI		
9			10	CH 25	CH 9 LO		
11			12	CH 18	CH 10 HI		
13			14	CH 26	CH 10 LO		
15			16	CH 19	CH 11 HI		
17			18	CH 27	CH 11 LO		
19			20	CH 20	CH 12 HI		
21			22	CH 28	CH 12 LO		
23			24	CH 21	CH 13 HI		
25			26	CH 29	CH 13 LO		
27			28	CH 22	CH 14 HI		
29			30	CH 30	CH 14 LO		
31			32	CH 23	CH 15 HI		
33			34	CH 31	CH 15 LO		
35	Analog Return	Analog Return	36	Not Used	Not Used		
37	Not Used	Not Used	38				
39			40				
41			42				
43			44				
45	Not Used	Not Used	46	Not Used	Not Used		
47	Analog Return	Analog Return	48	Analog Return	Not Used		
49	−15V	−15V	50	+15V	+15V		

B1

NOTE:

All odd-numbered pins (1, 3, . . . 49) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side with the board extractors at the top.

E-84

Table 2.15 Analog Input Connector J3 Pin Assignments

ORIGINAL PAGE IS
OF POOR QUALITY

BOARD - 2 - CURRENT

PIN	SINGLE-ENDED	DIFFERENTIAL	PIN	SINGLE-ENDED	DIFFERENTIAL	SENSE	MEASURED AT RELAY #
1	Not Used	Not Used	2	Not Used	Not Used		
3	Analog Return	Analog Return	4	CH 0	CH 0 HI	S4 +	
5			6	CH 8	CH 0 LO	S4 -	K4
7			8	CH 1	CH 1 HI	S1 -	
9			10	CH 9	CH 1 LO	S1 -	K1
11			12	CH 2	CH 2 HI	S2 +	
13			14	CH 10	CH 2 LO	S2 -	K2
15			16	CH 3	CH 3 HI	S5 +	
17			18	CH 11	CH 3 LO	S5 -	K5
19			20	CH 4	CH 4 HI	S3 +	
21			22	CH 12	CH 4 LO	S3 -	K3
23			24	CH 5	CH 5 HI	S6 +	
25			26	CH 13	CH 5 LO	S6 -	K6
27			28	CH 6	CH 6 HI	S11 +	
29			30	CH 14	CH 6 LO	S11 -	K11
31			32	CH 7	CH 7 HI	S9 +	
33			34	CH 15	CH 7 LO	S9 -	K9
35	Analog Return	Analog Return	36	Not Used	Not Used		
37	Not Used	Not Used	38	Not Used	Not Used		
39	Digital Common	Digital Common	40	Clock Out	Clock Out		
41			42	Ext. Trigger In	Ext. Trigger In		
43			44	EOC Status Out	EOC Status Out		
45	Digital Common	Digital Common	46	EOS Status Out	EOS Status Out		
47	Analog Return	Analog Return	48	Analog Return	Analog Return		
49	-15V	-15V	50	+15V	+15V		

NOTE:

All odd-numbered pins (1, 3, . . . 49) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side with the board extractors at the top.

+ IS THE SENSE MOTION CLOSEST TO THE POWER SOURCE

BOARD -2 - CURRENT

Table 2.14 Analog Input Connector J2 Pin Assignments

PIN	SINGLE-ENDED	DIFFERENTIAL	PIN	SINGLE-ENDED	DIFFERENTIAL	SENSE	MEASURED AT RELAY #
1	Not Used	Not Used	2	Not Used	Not Used		
3	Analog Return	Analog Return	4	CH 16	CH 8 HI	S10 +	
5			6	CH 24	CH 8 LO	S10 -	K10
7			8	CH 17	CH 9 HI	S7 +	
9			10	CH 25	CH 9 LO	S7 -	K7
11			12	CH 18	CH 10 HI	S8 +	
13			14	CH 26	CH 10 LO	S8 -	K8
15			16	CH 19	CH 11 HI	S13 +	
17			18	CH 27	CH 11 LO	S13 -	K13
19			20	CH 20	CH 12 HI	S14 +	
21			22	CH 28	CH 12 LO	S14 -	K14
23			24	CH 21	CH 13 HI	S12 +	
25			26	CH 29	CH 13 LO	S12 -	K12
27			28	CH 22	CH 14 HI	S18 +	
29			30	CH 30	CH 14 LO	S18 -	K18
31			32	CH 23	CH 15 HI	S16 +	
33			34	CH 31	CH 15 LO	S16 -	K16
35	Analog Return	Analog Return	36	Not Used	Not Used		
37	Not Used	Not Used	38				
39			40				
41			42				
43			44				
45	Not Used	Not Used	46	Not Used	Not Used		
47	Analog Return	Analog Return	48	Analog Return	Not Used		
49	-15V	-15V	50	+15V	+15V		

B2

NOTE:

All odd-numbered pins (1, 3, . . . 49) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side with the board extractors at the top.

E-95

Table 2.15 Analog Input Connector J3 Pin Assignments

BOARD - 3 - CURRENT

PIN	SINGLE-ENDED	DIFFERENTIAL	PIN	SINGLE-ENDED	DIFFERENTIAL	SENSOR	MEASURED AT RELAY #
1	Not Used	Not Used	2	Not Used	Not Used		
3	Analog Return	Analog Return	4	CH 0	CH 0 HI	S21+	K34
5			6	CH 8	CH 0 LO	S21-	
7			8	CH 1	CH 1 HI	S22+	BATTERY 1
9			10	CH 9	CH 1 LO	S22-	
11			12	CH 2	CH 2 HI	S23+	BATTERY 2
13			14	CH 10	CH 2 LO	S23-	
15			16	CH 3	CH 3 HI	S24+	BATTERY 3
17			18	CH 11	CH 3 LO	S24-	
19			20	CH 4	CH 4 HI	S19+	K31
21			22	CH 12	CH 4 LO	S19-	
23			24	CH 5	CH 5 HI	S15+	K15
25			26	CH 13	CH 5 LO	S15-	
27			28	CH 6 HI	CH 6 HI	S17+	K17
29			30	CH 14	CH 6 LO	S17-	
31			32	CH 7	CH 7 HI	S20+	BOARD 1
33			34	CH 15	CH 7 LO	S20-	
35	Analog Return	Analog Return	36	Not Used	Not Used		
37	Not Used	Not Used	38	Not Used	Not Used		
39	Digital Common	Digital Common	40	Clock Out	Clock Out		
41			42	Ext. Trigger In	Ext. Trigger In		
43			44	BOC Status Out	BOC Status Out		
45	Digital Common	Digital Common	46	EOS Status Out	EOS Status Out		
47	Analog Return	Analog Return	48	Analog Return	Analog Return		
49	-15V	-15V	50	+15V	+15V		

NOTE: + IS THE SENSOR POSITION CLOSEST TO THE POWER SOURCE

All odd-numbered pins (1, 3, . . . 49) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side with the board extractors at the top.

BOARD-3 - CURRENT

Table 2.14 Analog Input Connector J2 Pin Assignments

NC - NOT CONNECTED

PIN	SINGLE-ENDED	DIFFERENTIAL	PIN	SINGLE-ENDED	DIFFERENTIAL
1	Not Used	Not Used	2	Not Used	Not Used
3	Analog Return	Analog Return	4	CH 16	CH 8 HI
5			6	CH 24	CH 8 LO NC
7			8	CH 17	CH 9 HI
9			10	CH 25	CH 9 LO
11			12	CH 18	CH 10 HI
13			14	CH 26	CH 10 LO
15			16	CH 19	CH 11 HI
17			18	CH 27	CH 11 LO
19			20	CH 20	CH 12 HI
21			22	CH 28	CH 12 LO
23			24	CH 21	CH 13 HI
25			26	CH 29	CH 13 LO
27			28	CH 22	CH 14 HI
29			30	CH 30	CH 14 LO
31			32	CH 23	CH 15 HI
33			34	CH 31	CH 15 LO
35	Analog Return	Analog Return	36	Not Used	Not Used
37	Not Used	Not Used	38		
39			40		
41			42		
43			44		
45	Not Used	Not Used	46	Not Used	Not Used
47	Analog Return	Analog Return	48	Analog Return	Not Used
49	-15V	-15V	50	+15V	+15V

B3

NOTE:

All odd-numbered pins (1, 3, . . . 49) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side with the board extractors at the top.

E-86

Table 2.15 Analog Input Connector J3 Pin Assignments

ORIGINAL PAGE IS OF POOR QUALITY

A/D-0

(bytes)	Jumper
64K (16 bits)	none (no connection on 90, 91, 92, 93)
1M (20 bits)	90-91
16M (24 bits)	90-91 and 92-93

Ratio Jumpers

[redacted]

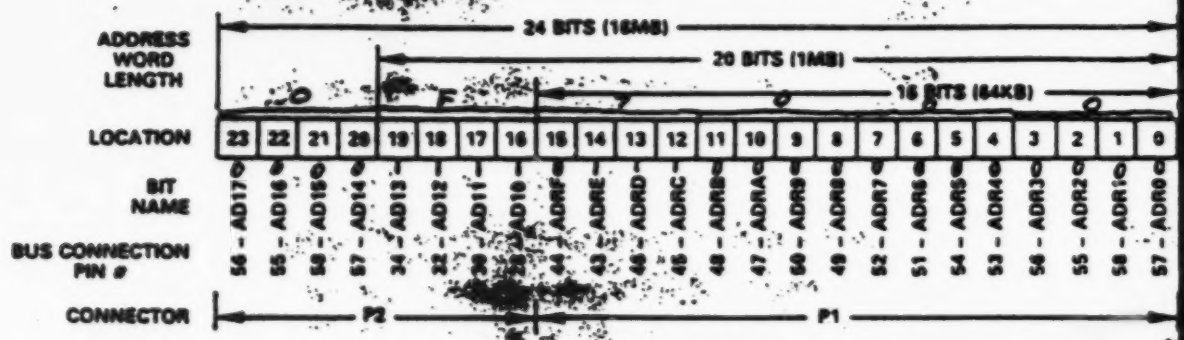


Figure 2.1 Memory Address Assignment

ADDRESS JUMPERS

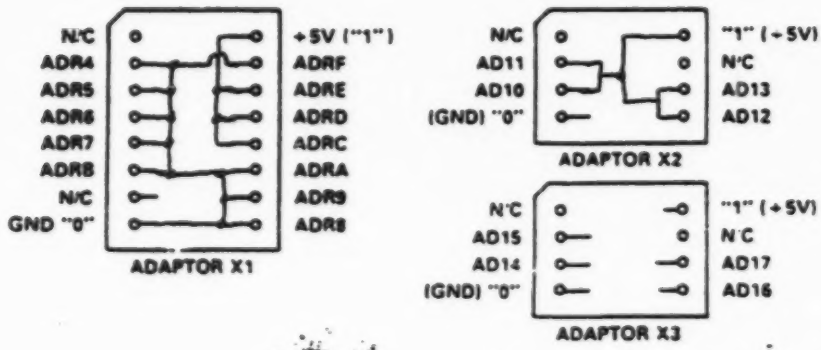


Figure 2.2 RTI-711-732 Jumper Blocks

2.9 MULTIPLEXER CONFIGURATION

The multiplexer can accommodate 16 single-ended or 8 differential input channels, and can be expanded to accommodate an additional 16 single-ended or 8 differential input channels as described in paragraph 2.17. The RTI-711-732 boards are shipped with jumpers configured for single-ended channel operation as shown in Figure 2.3a. If differential channel operation is [illegible] the jumpers as [illegible] shown in Figure 2.3b. The jumpers shown in Figure 2.3 are [illegible] in practice keep the jumpers as short as possible.

E-87

ORIGINAL PAGE IS OF POOR QUALITY

Addressable Memory Range (bytes)	Jumper
64K (16 bits)	none (no connection on 90, 91, 92, 93)
1M (20 bits)	90-91
16M (24 bits)	90-91 and 92-93

Table [illegible] Memory Jumpers

Example:

Figure 2.2 [illegible]

Note that each of the [illegible] ADR[illegible] jumpered either [illegible] (0) or +5V (1).

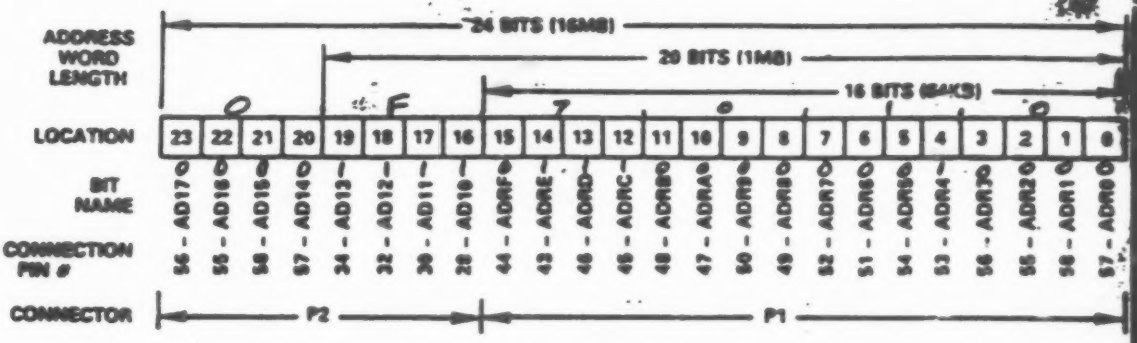
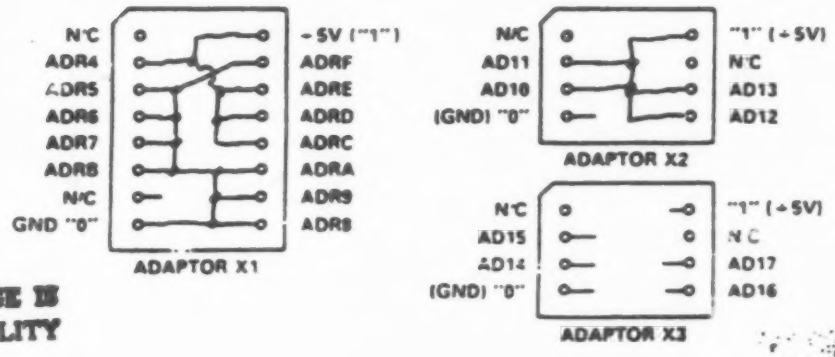


Figure 2.1 Memory Address Assignment



ORIGINAL PAGE IS OF POOR QUALITY

Figure 2.2 RTI-711 732 Jumper Blocks

2.9 MULTIPLEXER CONFIGURATION

The multiplexer can accommodate 16 single-ended or 8 differential input channels and can be expanded to accommodate an additional 16 single-ended or 8 differential input channels as described in paragraph 2.17. The RTI-711 732 boards are shipped with jumpers configured for single-ended channel operation as shown in Figure 2.3a. If differential channel operation is required, reconfigure the jumpers as illustrated in Figure 2.3b. The jumpers shown in Figure 2.3 are illustrated for clarity, in practice keep the jumpers as short as possible.

A/D-2

ORIGINAL PAGE IS
OF POOR QUALITY

Addressable Memory Range (bytes)	Jumper
64K (16 bits)	none (no connection on 90-91, 92-93)
1M (20 bits)	90-91
16M (24 bits)	90-91 and 92-93

Table 2.3 Memory Range Jumpers

Example:

Figure 2.2 shows how to configure the jumpers for a memory address of 35F700. Note that each of the address bits AD17-ADR4 must be jumpered either to GND (0) or +5V (1).

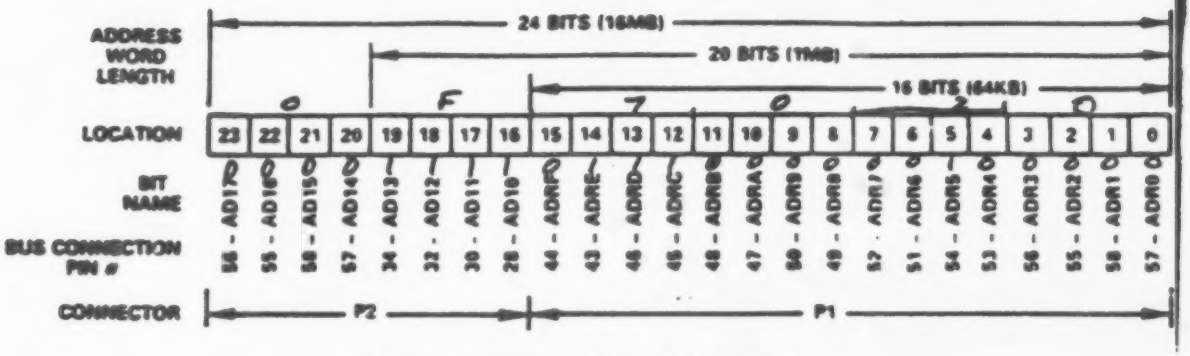


Figure 2.1 Memory Address Assignment

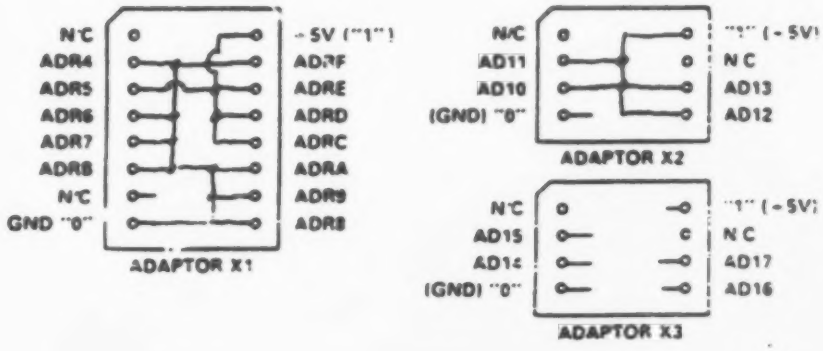


Figure 2.2 RTI-711/732 Jumper Blocks

2.9 MULTIPLEXER CONFIGURATION

The multiplexer can accommodate 16 single-ended or 8 differential input channels, and can be expanded to accommodate an additional 16 single-ended or 8 differential input channels as described in paragraph 2.17. The RTI-711/732 boards are shipped with jumpers configured for single-ended channel operation as shown in Figure [illegible]

E-89

A/D-3

ORIGINAL PAGE IS
OF POOR QUALITY

Addressable Memory Range (bytes)	Jumper
64K (16 bits)	none (no connection on 90-91, 92-93)
1M (20 bits)	90-91
16M (24 bits)	90-91 and 92-93

Table 2.3 Memory Range Jumpers

Example:

Figure 2.2 shows how to configure the jumpers for a memory address of 35F700. Note that each of the address bits AD17-ADR4 must be jumpered either to GND (0) or +5V (1).

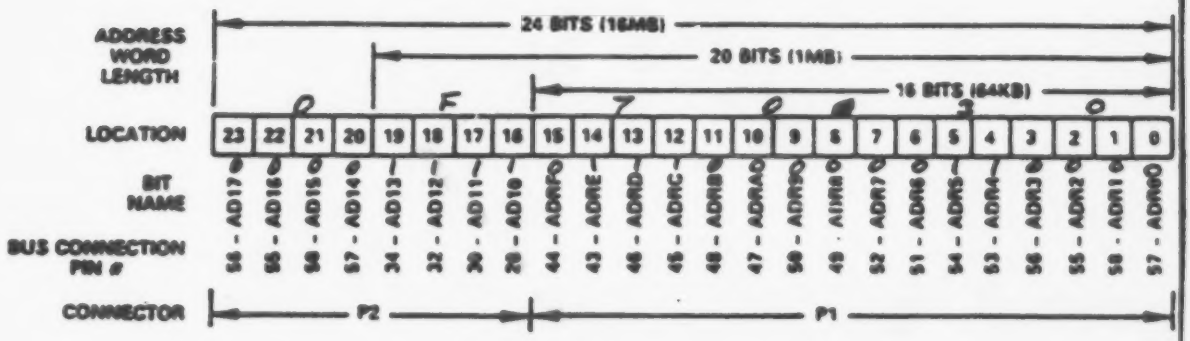


Figure 2.1 Memory Address Assignment

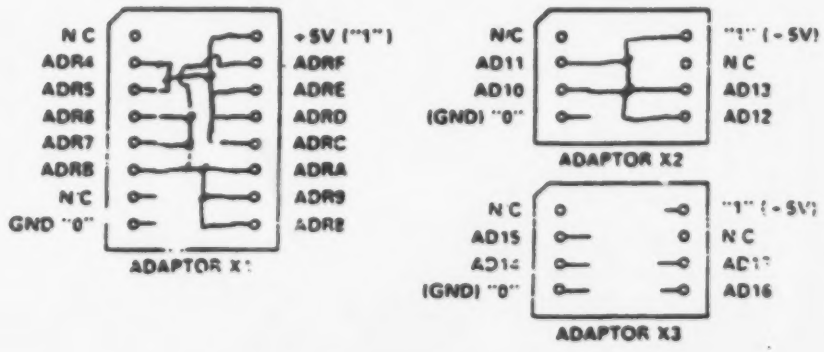


Figure 2.2 RTI-711 732 Jumper Blocks

2.9 MULTIPLEXER CONFIGURATION

The multiplexer can accommodate 16 single-ended or 8 differential input channels, and can be expanded to accommodate an additional 16 single-ended or 8 differential input channels as described in paragraph 2.17. The RTI-711 732 boards are shipped with jumpers configured for single-ended channel operation as shown in Figure [illegible]

E-90